IDA

INSTITUTE FOR DEFENSE ANALYSES

The Evolving Role of Analysis in Complex Military Decision-Making

Robert J. Graebener

Dr. Stephen Kasputis

March 2000

IDA Document D-2415

Log: H 99-002921

This work was conducted under contract DASW01 98 C 0067, Task DA-3-1697, for the Defense Advanced Research Projects Agency/ DISA JPO. The publication of this IDA document does not indicate endorsement by the Department of Defense, nor should the contents be construed as reflecting the official position of that Agency.

INSTITUTE FOR DEFENSE ANALYSES

IDA Document D-2415

The Evolving Role of Analysis in Complex Military Decision-Making

Robert J. Graebener

Dr. Stephen Kasputis

PREFACE

The Synthetic Theater of War (STOW) Advanced Concept Technology Demonstration (ACTD) concluded its full-life cycle with the operational user's (U. S. Joint Forces Command) acceptance of the simulation federation at the end of FY 99 and the successful transition of numerous technologies to various agencies and programs within DoD. The ACTD was jointly sponsored by the Defense Advanced Research Projects Agency (DARPA) and the USJFCOM.

The ACTD, while initially envisioned as vehicle for leveraging maturing technologies in support of the joint warfighter's mission, evolved into a capable simulation in its own right. During its short life cycle, USJFCOM required the technologies to support the following: Joint task force training program (Unified Endeavor 98-1); Analysis of ACTD's (Joint Counter Mine ACTD); and the first joint experiment (J9901, Critical Mobile Target, Attack Operations), set in the 2015 time period.

This report documents the complexities associated with current and future problem-solving techniques necessary to meet the military decision-maker's needs in the 21st century. The authors believe that the very nature of the STOW ACTD is a case study in what to expect from future challenges facing the warfighter, as well as the analytical community pledge to support joint commander. Three other reports document the evaluation of military utility, the technology, and an overarching assessment of the ACTD. This effort focuses on the scientific process and approach to complex problem solving necessary to meet tomorrow's challenges

This report is the result of collaboration between Mr. Robert Graebener of the Institute for Defense Analysis and Dr. Stephen Kasputis of VisiTech, Ltd. The authors acknowledge the contributions of several people for their significant contributions to this report. Mr. Rae Dehncke, PM STOW, for his willingness to give the authors the freedom to provide rigor and form to processes that were developed in parallel with technological initiatives supporting newly established joint warfighting missions. Dr. Andy Ceranowicz, for active feedback in the formulation of the process checklists. Messrs. Todd Morgan, Pete Wussow, Steve Haes, and Chuck Peters for providing value-added additions to specific functional procedures before and during the intensive events scheduled over the past 2 years.

Dr. Peter Brooks for comments provided in reviewing the manuscript. Mr. Tom Milani contributed substantially to the professionalism of the document through his editorial review. To the STOW team members, their professionalism, dedication and creative abilities provided the environment upon which to base these findings.

CONTENTS

FIGURES

TABLES

EXECUTIVE SUMMARY

The biggest challenge facing analysts today is the tradeoff between the need for detailed analysis and the timeliness requirements of the warfighter. If information from analyses is not available to the decision maker within his or her decision cycle time, it is of little or no value. The techniques and tools that support analysis must therefore provide the capability for the analysis to be completed within ever increasing time constraints.

While the dimensionality of the operational issue space continues to expand, the capability of the decision maker to consider multiple inputs remains unchanged. What this means for analysis is that matching the level at which the analysis is performed to the information required by the decision maker is more critical than ever. The techniques and tools used in an analysis must be precisely tuned to provide the information most critical to the decision maker. Too detailed information cannot be absorbed because of information saturation. Information that is too general will be of little use to the decision maker and will not contribute to lowering the risk associated with the decision.

Traditional approaches used in analysis must be augmented to meet the demands expected in the 21st century. The analytical community is at the portal of an era where advances in the enabling technologies make collaborative environments more accepted and more capable, reducing the delay times in providing critical information to the end user.

This report provides the reader with an awareness of the complexities associated with current and future problem-solving techniques necessary to meet the military decision makers' needs in the 21st century. The authors have used experiences gained through extensive work in the Synthetic Theater of War (STOW) Advanced Concept Technology Demonstration (ACTD). Three culminating events conducted during the ACTD lifecycle leveraged emerging simulation technologies developed by DARPA with emerging requirements for joint training, analysis, and experimentation expressed by the operational warfighter, U.S. Joint Forces Command. Using the traditional problem-solving approach as a starting point, this paper is the culmination of over 2 years' worth of hands-on assessment and critical thinking to answer the question,

> How can the military decision maker of tomorrow employ the latest technologies and analytical processes to make the best decision?

The answer lies in the following observations and recommendations:

- Establish standard procedures for the decision cycles.
- Ensure that all stakeholders are surveyed in the formulation of the purpose of the investigation.
- Identify long lead items early.
- Establish an early working relationship between the maintainers of the analysis tools and the stakeholders in the investigation.

The reader is cautioned not to conclude too hastily that these answers are readily known. Successful integration of complex concepts and support tools, along with the user and the organizational players—the means used to reach the ends—is the key.

It is the authors' belief that this report provides a way for the analytical community to move through the portal into the 21st Century. The goal is to present procedures for maximizing the potential for these methods, with emphasis on modeling and simulation, to support programs addressing complex issues.

1.0 INTRODUCTION

The biggest challenge facing analysts today is the tradeoff between the need for detailed analysis and the timeliness requirements of the warfighter. Timeliness requirements are being constantly made more stringent by the compression of decision lead times, the expansion of operational issue space relative to decision space, the increasing use of collaborative environments, and through advances in technology.

Lead times are shortening for all military decision makers. Tactical and strategic situations evolve faster than ever, decreasing the time allowed for the development of courses of action and contingency plans. Acquisition cycles are becoming ever shorter to ensure that new systems can be fielded before they become obsolete. Along with compression of decision cycles comes compression of the time available for the decision maker to gather the information required to make a decision. If information from analyses is not available within the decision maker's decision cycle time, it is of no value. The techniques and tools that support analysis must therefore allow the analysis to be completed within ever increasing time constraints. To do so requires continued improvement in the efficiency of the employment of analysis techniques and tools.

The operational issues facing any military decision maker are also expanding. In the tactical and strategic arenas, the dimensionality of the threat is increasing. In addition, the number of response options and the considerations for their employment continue to expand. The same is true for other types of military decisions. Acquisition program managers, for example, must consider not only increasingly complex combat systems, but also adherence to new regulatory legislation such as environmental protection and Occupational Health and Safety Administration (OSHA) standards. While the dimensionality of the operational issue space continues to expand, the capability of the decision maker to consider multiple inputs remains unchanged. Studies have shown that an individual is capable of effectively handling problems with up to seven different aspects. Beyond that, confusion begins to dominate, and problem solving efficiency decreases substantially. What this means for analysis is that matching the level at which the analysis is performed to the information requirements of the decision maker is more critical than ever. The techniques and tools used in an analysis must be precisely tuned to provide the most critical information. Too detailed information cannot

be absorbed because of information saturation; information that is too general cannot be further refined.

High-bandwidth communications, fast processors, improved databases, and the applications to tie them all together are leading to improved capability in, and therefore expanded use of, collaborative environments. Such environments provide near-instant access to the specific information required by each member of a team or project. The traditional computer-based tools of the analyst are now available to all the participants in the collaborative environment. Thus, the users watching the evolution of the simulation may expect answers directly from the simulation. The luxury of extensive off-line data processing can no longer be afforded. Any data processing that cannot be accomplished in near real-time will most likely be of limited value. This places a great burden on the simulation, the data collection mechanisms, the desired data processing, and the analyst.

The increasingly rigorous time demands placed on analysis tools and end products means that the efficiency of the analysis process is becoming more critically important. Not only the entire process used in complex problem solving, but also the separate procedures within that process, must be examined in the context of optimizing the overall process. Knowledge of these processes provides an understanding of the role of analysis within them. Insight into these processes also highlights the fact that coordination between the analysis team and the end users takes on added importance in ensuring that the experiments use the right tools properly tailored for the specific questions to be investigated and that the end user of the information has confidence that the answers are credible. A set of measures to assist in assessing the utility of analysis tools can speed the selection of the right tool for the specific investigation and thus improve the efficiency. Each of these aspects will be discussed in detail.

Central to the problem-solving process are the methods used to generate data for each investigation. For the purpose of this report, such methods will be classified as expert seminar, modeling and simulation, and live exercise. A working definition of and the distinction between these methods are detailed later in this report. It is important, however, to note that although there are different data-generation methods, they all have an important role to play in problem solving and the analysis that supports it.

In September of 1998, a pilot study was conducted in the U.S. Joint Forces Command (formerly United States Atlantic Command) using the simulation system of the Synthetic Theater of War (STOW) Advanced Concept Technology Development (ACTD). The goal of this study was to determine the value of entity-based simulation in supporting assessment of

the military utility of other ACTDs sponsored by the command. That is, the value of entity-based simulation as the data generation method for investigations of military utility was examined.

The central role of the data generation method in the investigative process afforded us the opportunity to explore the entire complex-problem-solving process. The goal was to determine approaches to achieve maximum efficiency. We studied the process, which used simulation as the data generation method, but most of the process, and therefore the approaches to achieve maximum efficiency, is independent of the data generation methods.

In the summer of 1999, a different type of analysis was begun with the support of the STOW simulation system. This study was the experiment, which served as a prototype process for Joint Experimentation. The purpose of Joint Experimentation differs from other forms of analyses. Rather than being a form of hypothesis testing such as typically done in evaluations or testing, the focus of Joint Experimentation is discovery of new operational concepts. For example, the attack operations joint experiment was designed to discover how emerging technologies and operational concepts might be used in the task of destroying land-based missiles.

In addition to addressing a different class of analysis, the experiments conducted during the summer of 1999 involved multiple players. The J9 of the Joint Forces Command (JFCOM) was the event director. The Joint Advanced Warfighting Program (JAWP) performed as the executive agent for the experiment. A second simulation was used in addition to the STOW simulation system. Successful execution of the experiment therefore required coordination among four major participants. This application of the STOW simulation system offered the opportunity for additional lessons learned from those of the previous uses of the STOW simulation system.

To fully understand the recommendations of this report, it is necessary to understand the context in which they were made. A framework applicable to the solution of complex problems is presented. This framework represents an iterative process in which certain procedures are repeated several times. These procedures are detailed and the critical or pacing operations highlighted. These procedures apply to any investigative cycle regardless of if the purpose of that cycle is hypothesis testing as part of complex problem solving or discovery. The recommendations emphasize identification and early addressing of the critical and pacing operations required for success of the investigation or experiment.

2.0 THE PROBLEM–SOLVING PROCESS

Complex issues require solutions at several levels of detail. That is, a general solution or approach is determined, and increasing levels of detail are then added. Generating such solutions is an iterative process, with successive iterations providing the increased detail. To illustrate, consider systems acquisition. The first stage of the acquisition process typically defines the most promising system concepts in terms of initial, broad objectives for cost, schedule, performance, requirements, opportunities for tradeoffs, overall acquisition strategy, and test and evaluation strategy. Subsequent program phases refine the concept and provide additional detail until a stable, interoperable, producible, supportable, and cost-effective design is obtained; validation of the manufacturing or production process is performed; and demonstration of system capabilities through testing is completed. Each refinement of the design is an iteration in the process.

Acquisition is but one functional area within DoD where such an iterative process is used to address complex issues. Iterative processes are used to evaluate concepts dealing with doctrine, force structure, organization, and funding priorities. Employment of iterative processes and the complexity of the iteration itself can vary greatly, depending on the problem addressed. Some problems require only a few iterations over a short period of time; some require hundreds of iterations over several years. Regardless of the purpose for which an iterative process is used or the extent of the iteration, however, one aspect is constant: within each of these iterations, postulated answers to one or more significant questions are investigated through some form of analysis to determine the "best" answer. As used here, "analysis" means the process of separating any material or abstract entity into its constituent elements as a method of studying its nature or determining its essential features and their relations. Since each iteration generally adds some detail, the analysis associated with each successive iteration will, in general, contain a more refined description of the constituent elements, essential features, and relations.

Figure 2-1 illustrates the concept of repeated application of basic investigative procedures to address complex problems. The spiral loops represent the basic investigative procedures applied numerous times within a program. Ensuring maximum efficiency of a process that involves repeated application of specific procedures suggests the need for a contextual framework for that process. There are multiple potential benefits from such a

framework. The structure can provide a template, which can be tailored to help ensure that all aspects of the investigative procedures are adequately addressed. As the procedures are tailored, the template also assists in developing an efficient investigation. Finally, the framework provides a context for estimating the resources required to conduct an investigation. In resource-constrained programs, this is a critical aspect to the design of affordable investigations.

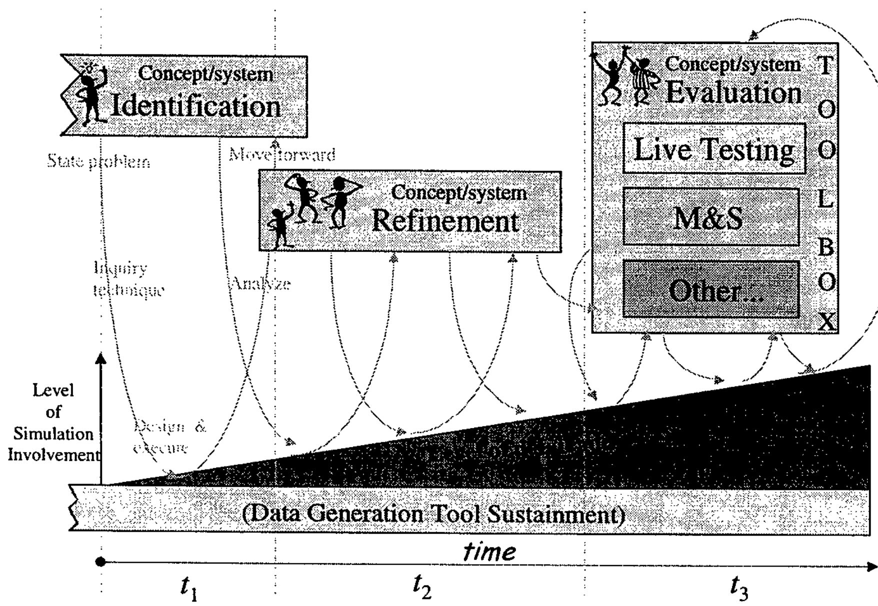

Figure 2-1. A Framework for Addressing Complex Problems

The output from each investigative iteration is information upon which the investigator can make decisions with respect to the questions posed. This requires that not only the correct information be produced, but also that the investigator understand and trust this information. Thus, the investigator must be involved in all aspects of the process, and all support personnel must be knowledgeable of the requirements of the investigation: the people executing the process are as important as the process itself, if not more so.

The basic procedures for each investigative iteration are fundamentally the same, although there are slight differences in emphasis between iterations focused on hypothesis testing and those focused on discovery. Because hypothesis testing accounts for the majority of the analyses conducted, it is discussed first. Differences necessary due to the nature of investigative iterations focused on discovery are then presented.

2.1 HYPOTHESIS TESTING

For each iterative cycle, the process must include identification of the question to be addressed, determination of the approach for gathering or generating the needed information, design and execution of the investigative procedures or experiment, analysis of the gathered information, and answering of the posed question to the extent possible.

The questions to be addressed can vary widely in subject area and level of detail, depending on the nature of the decision cycle that the iteration is supporting and the program stage. For example, a question for a program to develop new tactics could concern the optimum deployment formation of troops. The early stages of an acquisition program may be interested in refinement of top-level system requirements, while the pertinent question for a mature acquisition program may concern tradeoffs between two detailed design options.

The critical nature of properly and completely identifying the questions to be answered cannot be over-emphasized. Each and every stakeholder in the program must be surveyed. Not only must the fundamental questions of each stakeholder be identified, they must be identified to a sufficient level to allow determination of detailed data requirements. Definition of data requirements then provides the basis not only for the data collection plan, but also tools selection. These in turn are required for resource estimates for execution of the investigation and any modification to the tools used for the investigation. Along with resource estimates, identification of critical long-lead tasks and estimates for the time required for completion are also based on the identified data requirements.

Once the question to be investigated has been properly defined, MOEs that will support the answer can be developed. These measures identify the essential areas in which information must be collected to generate answers to the identified question. Weighting the MOEs by importance to the program is typically done, and this also provides a priority for data collection.

One additional activity that must be accomplished during the question identification phase of each iteration is delineation of any assumptions, both explicit and implicit. Some degree or level of assumption is unavoidable. These assumptions influence the defined question and the developed MOEs. Any final answers, decisions, or conclusions generated as a result of the iteration are dependent on these assumptions.

The next step before experiment design can begin is to identify the data and information requirements, which is obviously critical to the design of a proper and correct experiment. If MOEs have been defined, the required data are typically those needed to assess options against these MOEs. To support the subsequent efforts in event planning,

definition of the required data needs to be as specific as possible. The level of detail of data required is dictated by the questions. Initial investigations into the performance requirements of a proposed new weapon system may need such data as enemy losses over time as a function of sensor acquisition range. An investigation into which design option to pursue for a timing circuit for that weapon system's sensor would need more detailed data, such as sensor response times and required dwell times against specific targets.

Once the required data have been identified, options for generating the information needed to address them must be explored. As stated earlier, this report classifies these data-generation schemes as expert seminar, modeling and simulation, and live exercise. Expert seminar includes any and all methods that primarily involve generation or solicitation of expert opinion. Examples of this would include war game seminars such as map exercises and "rock drills," as well as top level and "back of the envelope" engineering analyses. Modeling and simulation includes exploitation of simple algorithms to calculate the battle-field results, use of estimation tools such as for network loading, and complex simulations that represent substantial portions of the battlespace. Live exercise includes activities as simple as the bench test of breadboard prototype components to large-scale field exercises.

Several methods of investigation could reasonably be classified under more than one of these techniques. For example, use of spreadsheets employing simple algorithms could be considered as modeling (or a back of the envelope engineering calculation). This report presents procedures for using these methods, with emphasis on modeling and simulation, to support programs addressing complex issues. Therefore, the simple classification scheme for investigative methods defined above is considered sufficient for this report. Selection of the data-generation method or methods best suited for a specific investigation requires consideration of several factors, which are discussed in detail in Section 3.0. Note that only rarely will the available data-generation methods provide exactly the data identified as required to address the questions of interest.

The next procedure within each iteration is experiment design and execution. The most important tasks in early experiment design are estimating the resources required and comparing the data-generation capability of the available investigative techniques with the defined data requirements. Because considerations in either area may well cause a modification to the questions being investigated, these are critical tasks.

Identification of resources may be as simple as confirming that the equipment normally in the workshop where development is taking place is sufficient for the bench test of a component breadboard. For large-scale field exercises, the required resources could

involve hundreds or thousands of troops, entire exercise ranges, and considerable quantities of expendables such as practice rounds, fuel, and food. Other examples include simultaneous availability of the desired experts for the seminar format or the required processing resources and operators for simulation events. Considerable attention should be given to the identification of required resources because some may require considerable lead time to obtain, but are nonetheless vital to event success. Also, care must be given that the resource requirements are estimated for all phases of the investigation.

Time must also be considered as a resource. Schedule estimates are required to ensure that the designed investigation can be conducted within the specified time frame. It is critical that long lead items be identified early enough that they can be properly addressed in the time allowed for the investigation. It is also important to assess the schedule of an iteration from a strategic perspective. That is, any one iteration must allow subsequent iterations to be completed in the time allowed for the final solution generation.

If the cost or time of the experiment as designed is too great, more than simple modifications of the experiment may be required. If the desired data cannot be economically obtained, the scope of the investigation may have to be changed. For example, modified questions to advance the program within available resources could be developed. Alternatively, the program might accept additional risk by basing the product of that iteration on less than optimal data. Either alternative would have further ramifications for the program as a whole.

Similar consequences can result from a comparison of data requirements with the data-generation capability of the methods selected for use in the investigation. In most instances, there will be some difference between the data that can realistically be collected and the ideal data required to fully answer the identified questions. In some cases, modification to the data generation and collection tools can dramatically reduce this difference, although it is unlikely to completely eliminate it. Therefore, as in the case of resource considerations, some modification of the questions of the investigation or expected quality of its results will probably be necessary.

Other issues must be considered during event planning. As mentioned earlier, some amount of adaptation of the selected techniques and tools for the specific experiment designed should be considered the norm. Included in this definition of adaptation is initialization. Adaptation in the areas physical representation, behavior, and interfaces as a minimum are required. Table 2-1 gives examples of possible adaptations for each of the investigative techniques discussed above.

Table 2-1. Example Adaptations

	Expert Seminar	Modeling and Simulation	Live Exercise
Physical Representation	Orient experts on initial force structures, capabilities, and deployments. Education on capabilities of new systems.	Addition of models of new systems to the established synthetic battle space.	Modification of signatures to emulate enemy systems. Addition of location transponders to units of interest. Modifications for safety.
Behavior	Education on proposed new doctrine.	Coding of new tactics, techniques, and procedures.	Education of units playing role of opposing forces on enemy tactics
Interfaces	Education of observers/recorders or facilitators. Establishment of the seminar rules of interaction.	Tailored interfaces to specific C4I systems. Modify filters of event data recorders.	Data tapes from a breadboard unit to the laboratory data recording devices.

Many investigations will be complex enough to require a formal event and data collection plan. This plan will include areas such as the development of scenarios, definition of initial conditions, staging of resources, and delineation of excursions to test the variability under specified conditions. Specifics on what data is collected when, by what means, and in what format is also typically addressed. A requirements trace matrix that relates specific data requirements to specific aspects of the question under investigation is usually included. Criteria for successful experiment runs and contingencies for specific types of failures may also be included.

Complex experiments may require considerable preparatory effort before actual execution. Participants or operators may need specialized training or education. If the experiment involves the coordination of several separate stages, rehearsal of some or all of the stages may be required. Often, such rehearsals result in some modifications to the event plan. In addition, all the logistics considerations identified earlier as necessary for event preparation need to put into place.

After proper planning and preparation, the experiment is conducted and data collected. If possible, the execution is compared against previously established success criteria to determine if a valid experimental run was completed. The data is formatted for analysis, and any required teardown of the experiment setup is conducted.

The final step in each iteration is the construction of a "product" appropriate to the question identified at the beginning of the iteration. Product is a general term that implies whatever is required to further the program's progress; however, it is generally some sort of decision supported by an analysis of the data generated by the investigation. Examples include system design decisions on any level, determination of optimum weapons mix in a given situation, or a prioritization of investments. The data is analyzed to determine the performance of the options available against the weighted MOEs derived earlier. If data requirements were properly defined and the experiment was successful at collecting this data, the initial questions can be answered, a decision can be made, and the program can move forward.

The decision represents the culmination of the iteration. It allows progression to the next identified question in the logical progression of program execution so the next iteration can begin. In this context, the analysis and accompanying decision are equivalent to the after action review of a training exercise. The analysis at the end of each iteration should make a strategic assessment of how the solution of the complex problem is progressing. The decision produced should thus be one that best directs the next iteration to contribute to the solution. Additional by-products of this effort are a refinement of the process used and improved estimates for the time and resources needed for the next event in the problem solving process.

2.2 DISCOVERY

Discovery uses a different form of analysis. The purpose of discovery is to improve the understanding of the nature of objects, events, and relationships. That is, insight into how things work is sought. This is significantly different than hypothesis testing, which typically asks only if things do work. Another way of thinking about the difference between the two types of analyses is that discovery essentially asks the question "Why?" and hypothesis testing asks the question "What?" Because of the different nature of such analyses, some changes are necessary to the execution of portions of the investigative iterations.

In discovery, definition of the purpose of the search replaces the identification of the question to be investigated, but it possesses the same critical nature. That is, the same care must be taken to ensure that all stakeholders are identified and their concerns considered. The same dependencies of identification of data requirements, tool selection, and identification of long lead items pertain. Because of the vagaries of the discovery process, starting with as clear an understanding of the purpose of the experiments is perhaps even more important for such analyses than for hypothesis testing.

Once the purpose of the search is defined, construction of MOEs needs to be performed. Instead of identifying the data required to answer posed questions, these measures identify the essential areas in which information must be collected to support the intended areas of discovery. As in the case of hypothesis testing, the MOEs should be prioritized. Definition of the MOEs similarly helps to focus the identification of the data required to support the study. Following identification of the required data, selection of the most appropriate tools for the experiment proceeds in the same manner as for hypothesis testing.

As with hypothesis testing, delineation of assumptions is an important step. Accurate identification of the assumptions and their possible consequences is, however, even more critical for discovery because discovery is an investigation of the nature of relationships, not simply the results of them. The nature of relationships is more complex than the results of them; their representation is more sensitive to changes in conditions or variables. Assumptions therefore have the potential to more greatly affect observation and perception of these relationships. Definition of the nature of relationships must be done in the context of the assumptions and their potential effects.

Experiment design and identification of resources required is similar to that done for hypothesis testing and has the same role in identifying limitations on end product. There is one significant difference in the consideration of required resources, however. For hypothesis testing, schedule is a derived requirement based upon the type and amount of data needed to address the questions posed. If the derived schedule requirement is too great, modifications to the questions and the scope of the investigation may be needed. For discovery, schedules can more correctly be characterized as constraints on the experiment rather than an estimate of a requirement. True discoveries—and developing the insight that leads to an understanding of the nature of things—do not follow a predictable schedule. Therefore, estimating when a discovery will occur is not realistic. For experiments focused on discovery, the aspect that requires adjustment is typically the expectation of the extent of the discovery or improved understanding of the nature of things. Once available resources are provided as inputs or constraints, expectation of the amount, nature, or detail of discovery can be estimated. Realistic estimates are not derived from simple equations, and generating them requires significant experience.

The other issues associated with each iteration as discussed under hypothesis testing are also areas of concern for discovery. Modification and proper initialization of tools will most likely be required. Formation of formal event and data collection plans that address the aspects previously discussed may be necessary. As with the hypothesis testing, a preparatory phase and following processes may be required. The final stage in the iteration is again the

production of an appropriate product. Unlike hypothesis testing, where this product is most often a decision, in analysis for the purpose of discovery, the product is typically an improved description of the nature of the objects, events, or relationships investigated.

Appendix A gives a sample detailed list of procedures for one decision iteration. This list can be tailored to accommodate decision iterations that are simple or complex and in any stage of the problem solving process. While presented for a specific application, the construction of this list was made to be as general as possible and thus maximize its potential applicability. In doing so, it was assumed that more than one organization would be involved in the definition, design, and conduct of the experiment and the interpretation of the results. In such instances, the steps of the list dealing with inter-organization coordination cannot be overemphasized.

The most appropriate way to characterize coordination between organizations depends on the relationship, their roles in the investigation, and the nature of the question to be addressed. Two obvious but useful such characterizations are discussed in Appendix B. These characterizations attempt to take into account the need for each for each organization to ensure that its interests are affirmed. Just as important, they also address means to ensure that each organization is comfortable with the manner in which their interests are addressed through program reviews or other appropriate interaction.

3.0 DATA GENERATION METHOD UTILITY MEASURES

The iterative process of solving complex problems begins with addressing questions at the most abstract level for which there are unknowns. As the process continues, the questions become more detailed as the solution becomes more defined. The process continues until a level of detail is reached that allows for a final resolution of the initial complex problem. This detail may represent the manufacturing plan of a new weapon system or the level of description necessary to promulgate new tactics, techniques, and procedures. Given this basic process, some requirements for the techniques and tools or collection of tools that support it are evident.

The first requirement is that the techniques employed must be able to address the entire problem space. The techniques and tools employed must be capable of assessing problems that span the extent of military operations under all possible physical and political conditions. All possible military or civilian systems that could be encountered and the range of behaviors or means in which they might be employed must be accommodated.

A major requirement for techniques supporting the investigative process is that they provide consistent results throughout the program. The tools that assist in the analysis conducted within each investigative iteration must be capable of assessing the elements, features, and relations at the correct level of detail to address the specific questions of that iteration. The level of detail of the analysis at each successive iteration differs from that of the previous iteration. To provide the consistency required throughout the process, either the tools that support the analysis must be capable of addressing a broad range of detail, or a suite of tools must be available that meets this requirement. It is unlikely that a collection of separate tools could meet this requirement without specific effort to integrate them into a consistent suite.

Fundamental to the employment of any method or tool is the confidence of the user that it will provide useful results. The results may indeed have rather strict validation limits. The user must be aware of these limits, however, and know how to employ the results in a valid manner.

Another requirement for the techniques employed to assist in developing solutions to complex problems is that they produce results that are in some sense repeatable. This does

not mean, for example, that a simulation needs to evolve in exactly the same manner each time it is run with a given set of initial conditions. It could mean for this simulation, however, that different sets of multiple runs of this simulation provide reasonably similar distributions of results. Without repeatability, there is no assurance that the decisions supported by a given technique will be the proper ones.

The type and scope of complex problems addressed by the iterative approach can vary greatly. It is therefore important that any techniques employed to support the process be adaptable. While they should be capable of handling investigations of large scope, they should also be capable of being employed in very specific investigations without the overhead that that may be required for expansive investigations. In this sense they must be scalable. They must also be adaptable in providing the level of fidelity affordable in terms of both time and resources to any given program. In short, they must be capable of providing a "quick and dirty" answer if that is all that is desired or can be afforded, and they also must be capable of providing more precise answers, if resources allow and the investigation demands it.

The ability to control the variables of the problem is important to identifying specific cause and effect relations. The tools used in an investigation must, therefore, afford the experimenter an appropriate level of control over the variables of the experiment. Available aspects of control may become important to the manner in which the experiment is designed and conducted. For example, simulations that provide the additional control of faster than real time execution may allow more flexibility in event design. This may allow for options such as conducting additional runs to investigate supplemental variations within the available time frame.

Each investigative iteration has a schedule for completion. The methods employed within each iteration must therefore support this schedule. It must be possible to assemble the required resources; set up, initialize, and run the experiment; analyze the data; and reach some conclusion within the allotted time frame.

Finally, the methods and tools employed in the investigative process must be affordable. The cost to execute, operate, and maintain the method or tool and to educate those who operate or use it needs to be considered. Areas of execution cost include not only the facility and operator costs associated with carrying out the experiment, but also the costs associated with preparation, post experiment cleanup, and data analysis. Preparation costs could include travel expenses for an expert seminar, software development for a simulation, or staging of forces for a live exercise. Cleanup costs could include shipment of borrowed

equipment back to the point of origin. The cost of data analysis is meant to include any data formatting or post processing and report generation as well as the analysis itself.

For the specific tools used to support the investigative process, the additional logistic areas of reliability and availability must be considered. That is, the level of effort required in an investigation can be greatly affected by the reliability and availability of the tools that support it. For example, if a simulation has poor reliability, many operator hours could be wasted either waiting or trying to recover from repeated faults. Although the exact requirements in these areas depend on the situation, they should always be considered in determining the appropriateness of tools.

Based upon the above requirements, some MOEs for the analysis support methods and tools can be created. The weighting and quantification of each MOE is dependent upon the specific application. In addition, because these MOEs are generally applicable, they represent only a very high level of consideration. Each can be further decomposed into multiple, more refined or detailed MOEs. Table 3-1 is a partial example of such a decomposition.

Table 3-1. Top Level MOEs and Sample Decompositions

Top level MOE	Sample Decomposition
To what extent can the method represent all the military operations needed throughout the iterative process?	Can all military and civilian entities be represented?
	Can all operational conditions be represented?
	Can all physical conditions be represented?
To what extent can the method provide data at any required level of detail?	Can variable level of detail be represented?
	Can detail be varied on a continuous or discrete manner?
	How many discrete level of detail are available?
	Can multiple level of detail be represented simultaneously?
To what extent are the limits of result defined?	Are the limits of the results known for every input variable?
	With what confidence are the limits known?
	What are the characteristics of the validity of the results near the limits?

continued

Table 3.1 (continued)

Top level MOE	Sample Decomposition
To what is extent is the applicability of the results defined?	Are the limits described in an operationally or analytically meaningful manner?
	What are the effects on the analysis if the results are applied to situations beyond the limits of applicability?
How repeatable are the results produced?	Are results repeatable at all levels of detail supported by the method?
	Are the results repeatable in an absolute manner?
	Are the results repeatable in a statistical manner?
What is the scalability of the method?	Can the method support investigations with different levels of resolution or scope?
	Is the level of resolution well defined and will it support are required iterations?
To what extent can the method produce only the required fidelity?	Is the method capable of producing results of varying levels of fidelity?
	Is the fidelity easy to control?
	Are the limits of applicability known for results at all levels of fidelity?
To what extent can the method isolate cause/effect relations?	Can the method ensure that only the desired variables are changed when and to the degree desired?
Can the method support the schedule requirements of the program?	Can required modifications be made in the time allowed?
	Can results be generated in the required time frame?
What is the cost of employing the method?	Is the cost of each phase of employment well known?
	How does the cost vary with changes in investigative scope, fidelity, or number of excursions?
	Are the estimated costs within the allowable range for the program?
What is the availability of the tool?	
What is the reliability of the tool?	

Appendix C provides a discussion of these measures specifically applied to the use of simulation as the data-generation tool in the context of complex problem solving. Appendix D gives an evaluation of the STOW simulation system. This evaluation does more than simply provide an assessment of the utility of the STOW simulation system. It provides an

example evaluation applicable to any type of data-generation techniques and illustrates how complex the utility measures can be.

4.0 RECOMMENDATIONS

To meet the increasing demands for timely decisions supported by analysis, the investigative process of identifying the critical questions to be answered, designing and executing an experiment to produce the data needed to answer those questions, analyzing the data, and developing a decision must be as efficient as possible. Toward achieving this goal, we recommend the following:

Establish standard procedures for the decision cycles.

These procedures need to be complete enough to address all aspects of each decision cycle throughout the problem-solving process. At the same time, the procedures should be flexible enough that they can be tailored to the different needs of specific investigations addressing different levels of detail at different stages of the problem-solving process. Appendix A contains an example set of such procedures. Appendix B discusses the coordination issues that arise when two or more organizations are involved in the definition and execution of these procedures.

Ensure that all stakeholders are surveyed in the formulation of the purpose of the investigation.

It is important that all questions of the investigation be identified before any experiment planning. Selection of the best method or methods to be used in data generation requires that all data requirements be identified. These requirements follow from the questions that are to be investigated. Identification of the long-lead items follows from the selection of the data-generation method and so is also highly dependent upon thorough knowledge of the goals of the investigation. The danger of not fully defining the questions to be investigated early in the decision cycle is that the efforts of the cycle will fail to produce the required decision. With late identification of questions of the investigation, the possibility exists that the data-generation technique already selected may not be capable of producing the data necessary to answer those questions. Another possibility is such questions might require long-lead items. If these long-lead items cannot be obtained within the time required for the decision cycle, either those questions must go unanswered, or the schedule must be adjusted to allow for a longer decision cycle.

Note that the identification of stakeholders will lead to requirements centered on the decision-making level of each interested party. Data collection requirements will subsequently have to be structured to address each level. This does not necessarily mean acquiring more resources to address each stakeholder issue, but it does mean that the end user will have to allocate collection efforts to balance the practical needs of the analysis against the political needs of the program.

Identify long-lead items early.

Long-lead items determine how quickly a decision cycle can be completed. The earlier such items can be identified, the faster the cycle can be completed. If long lead items are identified too late in the decision cycle, the emphasis of that cycle may need to be changed or a decision made on the data available. In such cases, the structure of the problem solution process would need to be adjusted accordingly. It is also possible that the lead time identified for items required for the employment of one tool type may be greater than the schedule allocated for the iteration. In such cases, the use of other tools needs to be considered along with the impact on the scope of the questions identified for that iteration.

Establish an early working relationship between the maintainers of the analysis tools and the stakeholders in the investigation.

Many of the advanced data-generation techniques are quite complex. They must be maintained and operated by specialists. Although these specialists are typically very good at the maintenance and operation of that specific analysis tool, they rarely have extensive knowledge of the questions to be investigated in the context of a particular problem. Likewise, because of the complexity of the data-generation methods, those familiar with the questions and purpose of the investigation are not likely to possess a detailed understanding of the capabilities of those methods. Knowledge exchange between these two groups is essential to making determinations. For example, if the data-generation method provides the data required, what long-lead tasks need to be accomplished, and what alteration to the tools may be necessary. Without adequate information exchange between these two groups, either the wrong data-generation method could be employed, the wrong data collected, or the wrong questions answered. For simulations requiring human-in-the-loop (HITL) interfaces, particularly in the manipulation of opposing forces and blue forces, the personnel selected normally are an excellent translation resource to ensure that the military terminology is understood by the technologist/developer.

ACRONYMS

AARS	After Action Review System
ACTD	Advanced Concept Technology Demonstration
ATM	asychronous transfer method
C4I	command, control, communications, computers, and intelligence
CCB	configuration control board
CCSIL	Command and Control Simulation Interface Language
COE	common operating environment
COMSEC	communications security
DARPA	Defense Advanced Research Projects Agency
DB	database
DEM	distributed exercise management
DII	Defense Information Infrastructure
DIS	Distributed Interactive Simulation
DTO	Dynamic Terrain and Objects
DVW	Dynamic Virtual Worlds
EA	executive agent
ECP	event control plan
EDB	environmental database
EPA	Environmental Protection Agency
ERC	experiment run check list
F&RDT	failure and recovery decision tree
FOM	Federation Object Model
FOTT	Federation Operations and Training Team
GCCS	Global Command and Control System
GPS	Global Positioning System
GUI	graphical user interface
HITL	human-in-the-loop
HLA	High Level Architecture
IC	Initialization Checklist
IR	infrared
JAWP	Joint Advanced Warfighting Program

JBC	Joint Battle Center
JFCOM	Joint Forces Command
JPO	Joint Project Office
JSAF	Joint Semi-Automated Forces
JSIMS	Joint Simulation System
JTASC	Joint Training, Analysis, and Simulation Center
KA	knowledge acquisition
LAN	local area network
LOE	level of effort
MAPEX	map exercise
METOC	meterological and oceanic
MOE	measure of effectiveness
MOP	measures of performance
NBC	nuclear, chemical, and biological
NRL	Naval Research Laboratory
OLE	Object Linking and Embedding
OPCON	Operational Control
OPFOR	opposing force
OSHA	Occupational Safety and Health Administration
PVD	plain view display
RPR FOM	Real-time, Platform Reference Federation Object Model
RTI	Run-Time Infrastructure
SAF	semi-automated forces
SEL	scenario event list
SLOGGER	STOW logger
SOP	standard operating procedure
SPAWAR	Space and Naval Warfare Systems Command
SPLAT	STOW performance logging and anomaly tracking
SSC	SPAWAR System Center
STOW	Synthetic Theater of War
SWA	Southwest Asia
TAOS	Total Atmosphere Ocean Services
TEC	Topographic Engineering Center
UJTL	Uniform Joint Task List
V&V	verification and validation
VV&A	verification, validation, and accreditation

WAN	wide-area network
WBS	work breakdown structure
WISSARD	What-If Simulation System for Advanced Research and Development
WWTDB	worldwide terrain data base

APPENDIX A—EXAMPLE DECISION CYCLE PROCEDURES

APPENDIX A—EXAMPLE DECISION CYCLE PROCEDURES

- Phase I - Joint Attack Ops and Support Tool Introduction
  - Initiate Exploratory Discussions
    - Joint Atk Ops Concept Development
      - Conduct Conceptual Wargame
      - Establish Operational Concept
      - Establish Base and Excursion Case Concept
      - Define Key Operational Issues
      - Define Operational Measures of Measurement
      - Establish Assumptions and Excursions
      - Establish Operational Scenario Concept
      - Formulate Analysis Plan Concept
    - Receive Concept Approval
    - Formally Introduce Joint Atk Ops
      - Describe Concept of Operations
        - "Classify Jt Atk Ops (e.g., future concept, test, etc)"
      - Identify special terrain and environmental requirements
      - Define Customer Test Issues
      - Identify Security Requirements
        - Determine Highest Level to Produce Valid Results
        - Assess Data Collection Requirements Impact on Security
    - Introduce JFCOM Intent (JFCOM Pay-off Areas):
      - Identify Joint Area(s) of Interest
      - Identify JFCOM Test Objectives
    - "Identify 'other' stakeholders and interests"
    - "Potential Leverages for Other Agencies (JBC, ACTD, Doctrine, Training)"
    - Establish Points of Contact Listing
  - Provide Jt Atk Ops Milestone Schedule
  - Assess Issues/Objectives
    - Utilize Process Alpha (Data Availability and Accuracy)
      - Begin Objective Decomposition into Component Parts
      - Identify Approaches Available to Address Questions (Issues)
      - Identify Data Required to Support Approaches
      - Assess Quality/Availability of Data to Support Objectives
      - Prioritize Objectives (Questions and Data Collection Focus)
  - Evaluate tools (methods) available for support
    - Inventory and Review Categories Available for support
      - A - STOW
      - B - Live Testing
      - C - Seminar
      - D - Other Simulations…
    - Develop user familiarity with categories and capabilities
    - Cross-familiarize customer/JTASC capabilities and requirements
  - Review Assessment Collection Methods
  - "Evaluate ""goodness of fit"" of Categories to support experiment "
  - Select category or categories necessary to support Jt Atk Ops objectives
- Phase II - Integration
  - Describe Jt Atk Ops Characteristics (High Level)
  - Phase II A - STOW Integration
    - Assess STOW Support Capabilities
    - Determine new work level of effort
      - Develop Physical Model(s)
        - Identify Requirements
          - Review Data Requirements
          - Compare to existing knowledge/Acquisition database
          - Determine KA Delta
        - Assess LOE Requirements
        - Modify Existing Physical Model(s)
          - Identify Surrogate STOW Physical Models
          - Utilize Process Bravo (Physical Model Development)
            - Develop Specific Knowledge/Acquisition (KA) Parameters

- Collect Data
- Modify Parameter Files
- Verify Code Operation
  - Test Against KA Description
    - Observe
    - Report Anomaly (ies)
    - Diagnose Cause
    - Cross Reference with Standard
  - Modify software to acceptable levels of representation
    - Repair
    - Bench Test (Static Analysis)
    - Network Test (Dynamic Observation)
- Validate object with respect to requirements
- Add New Physical Model(s)
  - Utilize Process Bravo (Physical Model Development)
    - Develop Knowledge/Acquisition (KA) Product
    - Cross-Level Model and KA Understanding
    - Compare to existing object database
    - Refine KA
    - Deliver KA Product
    - Develop Code
    - Verify Code Operation
      - Test Against KA Description
        - Observe
        - Report Anomaly (ies)
        - Diagnose Cause
        - Cross Reference w/ Standard
      - Modify software to acceptable levels of representation
        - Repair
        - Bench Test (Static Analysis)
        - Network Test (Dynamic Observation)
      - Validate Physical Model with respect to requirements
- Develop Behavior(s)
  - Identify Requirements
  - Assess LOE Requirements
  - Develop Behavior Model (Low Level)
    - Identify Requirements
    - Assess LOE Requirements
    - Utilize Process Bravo (Behaviors Development)
      - Develop Knowledge/Acquisition (KA) Product
      - Cross-Level Model and KA Understanding
      - Compare to existing behaviors database
      - Refine KA Requirements
      - Deliver product to software developer
      - Code Initial KA Representation
      - Verify Code Operation
        - Test Against KA Description
          - Observe
          - Report Anomaly (ies)
          - Diagnose Cause
          - Cross Reference with Standard
        - Modify software to acceptable levels of representation
          - Repair
          - Bench Test (Static Analysis)
          - Network Test (Dynamic Observation)
        - Validate with respect to requirements
- Develop Behavior Model (High Level)
  - Identify Requirements
  - Assess LOE Requirements
  - Utilize Process Bravo (Behaviors Development)
    - Develop Knowledge/Acquisition (KA) Product
    - Cross-Level Model and KA Understanding
    - Compare to existing behaviors database
    - Refine KA Requirements
    - Deliver product to software developer
    - Code Initial KA Representation
    - Verify Code Operation
      - Test Against KA Description
        - Observe
        - Report Anomaly (ies)

- Diagnose Cause
- Cross Reference with Standard (Requirement)
- Modify software to acceptable levels of representation
  - Repair
  - Bench Test (Static Analysis)
  - Network Test (Dynamic Observation)
- Validate with respect to requirements
- Develop Environmental Requirements (New)
  - Utilize Process Charlie (Terrain/Environment Development)
    - Identify terrain and environmental requirements
      - Define Physical Properties
      - Establish Level of Detail
    - Develop Knowledge/Acquisition (KA) Product
    - Cross-Level Model and KA Understanding
    - Compare to existing Environments data base
    - Identify data requirements
    - Assess LOE Requirements
    - Search and extract data from authoritative data sources
      - Specify
      - Search
      - Access
      - Evaluate
      - Retrieve
  - Generate the environmental database (EDB)
    - Terrain and Bathymetry
      - Cleaning
      - Thinning
      - Spot Intensification
      - Re-Representation
    - 3-D Models
      - Objects
        - Simple
          - Utilize Process CC
            - Develop Object
            - Observe Functionality
            - Modify Software
            - Observe Functionality
            - Accept Object
        - Complex
          - Utilize Process CC
            - Develop Object
            - Observe Functionality
            - Modify Software
            - Observe Functionality
            - Accept Object
      - Effects
        - Simple
          - Develop Effect
          - Test
        - Complex
          - Develop Effect
          - Test
    - METOC
      - Utilize Archived Observation Data Sets
      - Generate New Data Sets
      - Capture Real World Data Sets
  - Compile EDB into Run-Time Data Base
  - Incorporate Warfighting Activity Effects
  - Interaction Verification
    - Verify Code Operation
      - Test against terrain and environmental requirements
        - Observe
        - Report Anomaly (ies)
        - Diagnose Cause
        - Cross Reference with Standard (Requirement)
      - Modify software to acceptable levels of representation
        - Repair
        - Bench Test (Static Analysis)
        - Network Test (Dynamic Observation)

        - Validate with respect to requirements
    - Develop C4I/HITL (Man-In-The-Loop)/Other Interfaces
      - Utilize Process Delta (C4I/HITL Interface Development)
        - Conduct Interface Concept Discussion
        - Develop C4I/HITL requirements
          - Define Protocol
          - Define Information Exchange Format
          - Define Interconnectivity Architecture
        - Identify P.O.C's for Each Side of Interface
        - Establish Configuration Management
          - Track Changes
          - Ensure Configuration Control
        - Cross level STOW C4I Current Capabilities
        - Develop initial interfaces
        - Verify Code Operation
          - Test against C4I and HITL requirements
            - Observe
            - Report Anomaly (ies)
            - Diagnose Cause
            - Cross Reference with Standard (Requirement)
          - Modify software to acceptable levels of representation
            - Repair
            - Bench Test (Static Analysis)
            - Network Test (Dynamic Observation)
        - Validate with respect to requirements
    - Develop Assessment Requirements
      - Utilize Process Echo (Assessment Plan Development)
        - Collect all Jt Atk Ops assessment objectives
        - Continue Objective Decomposition into Measurable Sub-Objectives
        - Identify Assessment Deliverable/Product
        - Define MOPs/MOEs
        - Develop Assessment Team Capabilities and Resource Level
        - Compare to assessment objectives
        - Validate with respect to requirements
      - Utilize Process Foxtrot (Data Collection Development)
        - Decompose Jt Atk Ops tactics/techniques/capabilities
        - Compare to MOPs/MOEs
        - Determine data targets within simulation
        - Identify Analysis Components
          - Observer/Controller (HITL Eval)
          - Sim Collection Agents (MOPs/MOEs)
          - Sim Collection Agents (Visual and Network Performance)
        - Develop AARS Collection Requirements
        - Develop code in support of collection requirements
        - Refine Assessment Team Capabilities and Resource Level
        - Verify Code Operation
          - Test data collection code
            - Observe
            - Report Anomaly (ies)
            - Diagnose Cause
            - Cross Reference with Standard (Requirement)
          - Modify software to acceptable levels of representation
            - Repair
            - Bench Test (Static Analysis)
            - Network Test (Dynamic Observation)
        - Validate with respect to requirements
      - Synchronize with Other Tool Assessment Programs
      - Present Assessment Team Resource Requirements
  - Develop integrated milestone schedule
  - Identify funding required to meet objectives
  - Contract for software modifications/improvements
- Phase II B - Live Testing Integration
  - (Separate process development effort required)
- Phase II C - Seminar Integration
  - (Separate process development effort required)
- Phase II D - Other Simulations...Integration
  - (Separate process development effort required)

- Phase III - J99-01 Preparation
    - Manage Jt Atk Ops Test Events
        - Review Simulation Support Plans
        - Conduct Trade-Off Analysis
        - Prioritize Simulations' Efforts
        - Integrate Simulation's Assessment Plans
    - Phase III A - STOW Supported J99-01 Preparation
        - Assign J99-01 manager
        - Develop J99-01 Control Plan (ECP)
        - Publish J99-01 Control Plan
        - Conduct Initial Concept Development of STOW-Jt Atk Ops event
            - Identify J99-01 Objectives
                - Evaluate C4I Interface Requirements
                - Evaluate Assessment Requirements
                - Identify Stage-By-Stage Analysis Process
                - Develop Data Collector Forms/Procedures
                - Assess New Work Impact
            - Identify Distributed Sites
                - Player Sites
                - Assessment Sites
            - STOW-Jt Atk Ops Conops Planning
            - Develop overall scenario framework
                - Establish Scenario Concept
                - Assess Additional Software Requirements to Support Scenario
            - Develop Jt Atk Ops system vignettes
                - Establish Vignette Concept
                - Develop Scenario Event List (SEL)
                    - Draft SEL
                    - Update SEL
                    - Finalize SEL
                - Develop Customized Failure & Recovery Decision Tree (F&RDT)
                    - Draft F&RDT
                    - Update F&RDT
                    - Finalize F&RDT
                - Assess Additional Software Requirements to Support Vignettes
                - Establish J99-01 Configuration Control Board
                    - Initiate J99-01 Bug Fix Reporting System (SPLAT)
                    - Collect and Review Bug Reports
                    - Determine bug fix priorities (tied to scenario)
                    - "Establish CCB leads from event, tech, ops and assessment"
                    - "Promulgate bug priorities for overall, and individual service attention"
                    - Coordinate Software Releases
            - Identify J99-01 Scenario terrain location
            - Develop Force structure for scenario
            - Prepare FOTT team testing Schedule
            - Develop Federation Operations and Training Team Support Requirements
            - Prepare Event Milestone Schedule
            - Draft VIP Visit Plan
                - Identify VIP Program Coordinator
            - Coordinate data library space for event output
            - Integrate vignettes into overall scenario
                - Develop Scenario and Vignette Data Bases
                - Establish Unique Identifier for ALL Entities
                - Create Scenario Files
                - Utilize Process Golf (Scenario File Development)
                    - Verify Code Operation
                        - Test Integration
                            - Observe
                            - Report Anomaly (ies)
                            - Diagnose Cause
                            - Cross Reference with Standard (Requirement)
                        - Modify software to acceptable levels of representation
                            - Repair
                            - Bench Test (Static Analysis)
                            - Network Test (Dynamic Observation)
                    - Validate Scenarios Support Requirements
            - Integrate Assessment Process and Tool with Scenario
                - Conduct Pilot System Integration Run
                    - Produce Multiple Run SLOGGER Tapes
                    - Load and Analyze Data within AARS

        - Utilize Process Hotel (Data Collection Tool Evaluation)
          - Verify Code Operation
          - Test AARS Products
            - Observe
            - Report Anomaly (ies)
            - Diagnose Cause
            - Cross Reference with Standard
        - Modify software to acceptable levels of representation
          - Repair
          - Bench Test (Static Analysis)
          - Network Test (Dynamic Observation)
    - Conduct Pilot System Crash Excursions
      - Produce Multiple Run SLOGGER Tapes
      - Load and Analyze Data within AARS
        - Utilize Process Hotel (Data Collection Tool Evaluation)
          - Verify Code Operation
          - Test Against KA Description
            - Observe
            - Report Anomaly (ies)
            - Diagnose Cause
            - Cross Reference with Standard
          - Modify software to acceptable levels of representation
            - Repair
            - Bench Test (Static Analysis)
            - Network Test (Dynamic Observation)
    - Validate Assessment Tool Products
    - Develop Test Plan
    - Distribute SOP's to respective personnel
    - Conduct MAPEX
- Determine J99-01 Support Requirements
    - Assess external sim support requirements (outside of JFCOM)
    - Coordinate (event) network accreditation of ATM
    - Update J99-01 Management personnel requirements
- Contract for external site support (distributed simulation support)
- Identify modes of communication
- Fill vacant spaces in EvMgmnt
- Schedule internal net and external ATM accreditation at each site
- Develop Experiment Training Support Program
    - Develop Operator Training Requirements
- Complete equipment agreements
- Execute Test Plan
    - Conduct Initial FOTT
      - Configure ATM connection
      - Conduct low-level connectivity test
      - Conduct Operator Training - Hands On
      - Conduct Operator Training - Test Environment Familiarization
      - Conduct Assessment Team Training - Hands On
      - Conduct After Action Review
        - Adjust Scenario Modifications
        - Adjust Data Collection and Analysis Procedures
        - Assess Software Improvements
    - Conduct Interim FOTT(s)
      - Utilize Process Golf (Scenario File Development)
        - Verify Code Operation
        - Test Integration
          - Observe
          - Report Anomaly (ies)
          - Diagnose Cause
          - Cross Reference with Standard (Requirement)
        - Modify software to acceptable levels of representation
          - Repair
          - Bench Test (Static Analysis)
          - Network Test (Dynamic Observation)
      - Conduct Operator Training - Hands On
      - Conduct Operator Training - Test Environment Familiarization
      - Conduct After Action Review
        - Adjust Scenario Modifications
        - Adjust Data Collection and Analysis Procedures
        - Assess Software Improvements
    - Conduct Final FOTT

Utilize Process Golf (Scenario File Development)
- Verify Code Operation
- Test Integration
  - Observe
  - Report Anomaly (ies)
  - Diagnose Cause
  - Cross Reference with Standard (Requirement)
- Modify software to acceptable levels of representation
  - Repair
  - Bench Test (Static Analysis)
  - Network Test (Dynamic Observation)

Utilize Process Hotel (Data Collection Tool Evaluation)
- Verify Code Operation
- Test AARS Products
  - Observe
  - Report Anomaly (ies)
  - Diagnose Cause
  - Cross Reference with Standard (Requirement)
- Modify software to acceptable levels of representation
  - Repair
  - Bench Test (Static Analysis)
  - Network Test (Dynamic Observation)

Conduct Operator Training - Hands On
Conduct Operator Training - Test Environment Familiarization
Conduct Assessment Team Training - Hands On
Conduct After Action Review
- Adjust Scenario Modifications
- Adjust Data Collection and Analysis Procedures
- Assess Software Improvements
- Assess Operator Training Deficiencies

Conduct System Accreditation
- Accredit Model for Experiment
- Accredit Assessment Tool for Experiment

Phase III B - Live Testing Supported Event Preparation
(Separate process development effort required)
Phase III C - Seminar Supported Event Preparation
(Separate process development effort required)
Phase III D - Other Simulations...Supported J99-01 Preparation
(Separate process development effort required)

Phase IV - J99-01 Execution (Current Evolution)
- Phase IV A - STOW J99-01 Execution (Multiple Scenario Runs)
- Initiate Scenario Run
  - Pre-Set Systems
    - Ensure Software Versions Synchronized and Operational
    - Ensure Networks Available and Operational
    - Perform Maintenance and Non-Emergency Configuration Changes
    - Reinforce Operators' Role in Experiment
  - Conduct Initialization Checklist (IC)
    - Poll Each Participating Site
    - Ensure DEM Operational
    - Ensure SLOGGER Operational
    - Declare Simulation System Cleared for Run
  - Follow Scenario Run Checklist (ERC)
    - Poll Each Participating Site
    - Declare Simulation System and Sites Ready
  - Initiate Scenario Run
- Scenario Run
  - Conduct Run-Time Assessment
    - Utilize Assessment Run Criteria and SEL
    - Recommend Acceptance/Non-Acceptance of Segment Run
  - Execute Scenario Management
    - Scenario Run
      - Track Progress thru Scenario Event List (SEL)
      - Execute Surrogate Staff Requirements
      - Report Progress Thru Scenario Event List (SEL)
      - Maintain Centralized Log
      - Conduct Scenario Run Analysis
      - O/C Input Review
      - Sim Data Sampling
    - Implement Recovery Procedures

- Observe
  - Report Anomaly (ies)
  - Diagnose Cause
  - Cross Reference with Standard (Requirement)
- Drill for Technical Failures
  - Report to Event Control ASAP
  - Ensure DEM Automatic Capture Capability In Progress
  - Sites Review Last Save Points
- Drill for JSAF Failures
  - Report to OPCON ASAP
    - Provide Assessment of JSAF Failure
    - Provide Recommendation for Re-Start/Recovery Point
  - Review Recommendation with respect to F&RDT Criteria
- Determine Failure Impact
  - No Impact to Overall Scenario Segment
    - Restart at Latest Save Point
    - Document Information for Assessment Team
  - Impact to Overall Scenario Segment
- Notify Assessment Team and Test Director
- Present Courses of Action
- Decision to Restart or Continue Scenario Run
- Assess Re-Start Recovery Point for All Sites
- Synchronize Re-Start with All Sites

Scenario Run Completion
- Complete Post-Event Checklist
  - Poll Each Site
  - Submit Site Summary Report
- Review Segment Compliance

Post-Scenario Run AARS Functions
- Populate AARS Repository w/ Logged Data
- Analysis Agents Run to Generate Analysis Products
- Review AARS File for Run Compliance/Anomalies

Decide on Acceptance/Re-run

Phase IV B - Live Testing Event Execution
- (Separate process development effort required)

Phase IV C - Seminar Event Execution
- (Separate process development effort required)

Phase IV D - Other Simulations... J99-01 Execution
- (Separate process development effort required)

Phase V - Post J99-01 Operations (Future Evolution Planning)
- Phase V A - STOW Post J99-01 Operations
  - Prepare AARS Data for Customer Analysis
  - Deliver AARS Products
  - Update SOP's
  - Review software upgrade procedures and decision authority
  - Conduct Experiment/J99-01 Support After Action Review
  - Re-Cycle Process to Phase I
- Phase V B - Live Testing Post Event Operations
  - (Separate process development effort required)
- Phase V C - Seminar Post Event Operations
  - (Separate process development effort required)
- Phase V D - Other Simulations... Post J99-01 Operations
  - (Separate process development effort required)
- Phase Independent (STOW Availability)
  - Network Connectivity
    - Plan EMC modifications (w/s on WAN)
    - Identify reserved IP assignments
    - Monitor COMSEC Code Change Cycle
    - Maintain overall network accreditation
  - Hardware
    - Maintain Inventory Records
    - Continue Infusion of New Technology
  - Personnel
    - Maintain Surge Hire Expertise Files
  - Training
    - Conduct Training Program
      - Army
      - Navy
      - Marines
      - Air Force

OPFOR
C4I

Planning
- Coordinate Long Range Requirements
- Conduct Cost Analysis
- Assessment Planning

Maintain Documentation
- Coordinate with JSIMS and JWARS Programs
- Update Initialization Check List

Maintain STOW Configuration Management
- Fix Latent Deficiencies
  - Support Bug Fix Reporting System (SPLATTER)
  - Collect and Review Bug Reports
  - Determine bug fix priorities (tied to scenario)
  - "Promulgate bug priorities for overall, and individual service attention"
- Utilize Process India (Software Maintenance)
  - Verify Code Operation
    - Test Integration
    - Observe
    - Report Anomaly (ies)
    - Diagnose Cause
    - Cross Reference with Standard (Requirement)
  - Modify software to acceptable levels of representation
    - Repair
    - Bench Test (Static Analysis)
    - Network Test (Dynamic Observation)
- Validate Scenarios Support Requirements
- Establish Software Release Cycle
- Software Release
- "Maintain File Updates with ""Centers of Excellence"""
  - TEC
  - SSC
  - NRL
  - JPO
  - WISSARD

APPENDIX B—INVESTIGATIVE CYCLE MANAGEMENT AND COORDINATION OF PARTICIPATING ORGANIZATIONS

APPENDIX B—INVESTIGATIVE CYCLE MANAGEMENT AND COORDINATION OF PARTICIPATING ORGANIZATIONS

Most iterative investigative cycles involved with solution generation for complex problems involve the efforts of two or more organizations. The techniques used to coordinate these organizations depend on their relationship and role in the investigation. Two different characterizations of the coordination effort are presented below. One is based on a hierarchy of responsibilities, and the other is based on the premise that each organization is doing an investigation and that all investigations follow basically the same process. An assumption on both characterizations is that all the organizations are interested in the same experiment or set of experiments.

In attempting to characterize the interactions between organizations, it is necessary to ensure that the characterizations address two critical aspects: (1) the division of responsibility between the organizations and (2) the authority to control the aspects for which an organization is responsible. Effective exercise of authority requires information to support the necessary decisions. If the overall collaborative effort is to be successful, the structure of the interactions among multiple organizations must allow each organization to acquire the information it needs to exercise its authority. Any characterization of the interactions between organizations must also address this aspect of the interactions by capturing the structure for program reviews and other technical exchanges. The characterizations presented below attempt to address both the division of responsibility among the organizations and the structure for information exchange necessary to execute an investigation.

B.1 HIERARCHICAL CHARACTERIZATION

The tasks associated with an investigation can be placed in a classic work breakdown structure (WBS) format. If participating organizations can be classified as seniors and subordinates within some chain of command, one instantiation of the hierarchical characterization is each organization being responsible for different levels of detail of the WBS. The sponsor or program manager is responsible for the highest levels, which establish the overall program structure and direction. The executive agent (EA) is responsible for the middle layers, which establish the overall experiment architecture. The tool maintainer or

executor is responsible for the lowest layers of the WBS, which contain the detailed tasks required to prepare, conduct, and analyze the experiment.

With such a way of looking at the problem, the areas of intersection can be viewed as either collaboration, defining the "interface" between levels of different organizational responsibility, or as joint responsibility for defining a level where responsibility transitions from one organization to another. Figures B-1 and B-2 illustrate these options. In Figure B-1, the sponsor organization is responsible for the highest two layers. The EA defines the next two layers of detail. The maintainers and executors of the tools to be used in the investigation/analysis must specify the remainder of the experiment detail. There must be a distinct and well understood definition of the lowest sponsor level of detail that provides unambiguous direction to the EA for the definition of the next layers of detail. There must be a similarly well understood and unambiguous definition of the lowest level of detail provided by the EA so that those who have the best knowledge of the analysis tool use can complete the detail required for an experiment. Obtaining the required clarity at these transitions levels will require collaboration between the organizations on either side of the interface.

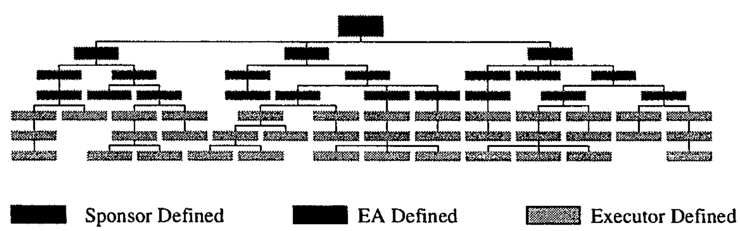

Figure B-1. Exclusive Work Breakdown Structure Responsibilities

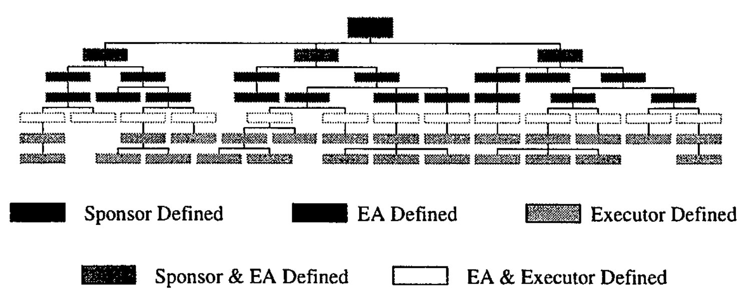

Figure B-2. Shared and Exclusive Work Breakdown Structure Responsibilities

Figure B-2 is an alternative view that emphasizes the collaboration between the different organizations. Each organization still maintains responsibility of some specific levels of detail. Where responsibility passes between organizations, however, is a level of

detail for which organizations share responsibility. The interface in this instance is embedded in that level of detail. This eliminates some of the rigor required for interface definitions.

In both instances, the interface in question is really an interface between organizations. The only way to make such an interface work is through frequent interactions. The framework of a WBS focuses the essential areas of interaction. Note also that there are fewer topics (WBS boxes) on which the sponsor and the executive need to interact than for the EA and the executors.

It is possible to have more than one executor organization. If elements of the WBS can be clearly and unambiguously divided between them, this will not appreciably increase the needed amount of coordination required for the project. It is more likely, however, that some of the WBS elements will require effort from multiple executors. This adds a lateral as well as vertical dimension to the needed coordination, increasing the oversight requirements at the EA level.

Such a relationship between sponsor, EA, and tool maintainer/executor also provides a framework for understanding program reviews by all parties. In general, the level of detail most appropriate for program reviews by any organization will be at the first level below its area of responsibility. There will always be specific issues from the lower levels of detail that rise to higher level program interest because of their critical nature; however, review of the actions and progress at the first level below the level of responsibility defined will give a manager an indication to what degree directions are being followed.

A different instantiation of the hierarchical structure is possible for organizations not in the same chain of command, but rather interested in questions of different levels of detail that can, nonetheless, be addressed in the same experiment or set of experiments. This could only occur for questions that lend themselves to logical decomposition. Such questions generally do not involve several interrelated variables. For such instances, a parallel to the WBS is a hierarchical breakdown of investigative detail. The division of responsibility between organizations could be similar to the senior-subordinate division discussed above. Each organization would be most responsible for the definition of the data required at the level of detail for which they have the greatest interest. The interface between levels would be the information fusion techniques used to aggregate from levels of greater to lesser detail. These fusion techniques are the areas where the coordination between organizations needs to concentrate. The division of responsibility for the defining these fusion techniques would determine if the organizational coordination could be better characterized by Figure B-3 or Figure B-4, analogous to Figures B-1 and B-2, respectively.

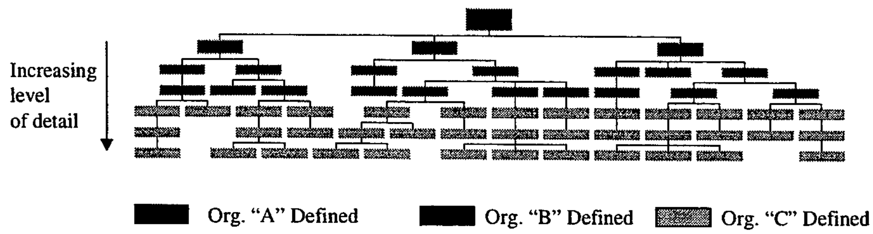

Figure B-3. Exclusive Data Requirement Definition Responsibilities

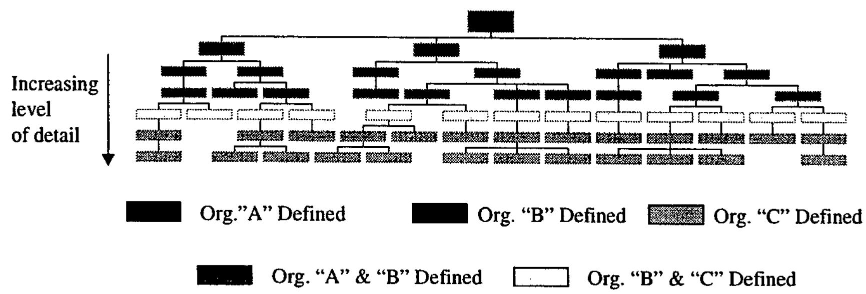

Figure B-4. Shared and Exclusive Data Requirement Definition Responsibilities

B.2 SIMILARITIES IN PROCESS

Another approach to viewing the coordination of the investigative process among several organizations is based upon the similarities of all investigations, even though each organization may represent the basic steps differently or use different semantics in describing them. The basic procedures for each investigation were previously established. The question to be addressed is identified, the approach for gathering or generating the needed information is determined, the investigative procedures or experiment is designed and executed, and the gathered information is analyzed and the question answered to the extent possible. These basic steps apply to all stakeholders, even though they may delegate aspects of the event design or execution. Because delegation of authority does not relieve that organization of responsibility, it must have confidence in whatever investigation or experiment is conducted. In coordinating processes based upon these inherent similarities, a high-level timeline version of each process is presented. The next step is to identify the points where collaboration should logically be expected or where it should benefit the common interests of the organizations.

The points of intersection of these processes depend on the relationship of the organizations. Such a relationship can be of a senior-junior or customer-provider nature as

discussed above, or as among equals or contemporaries. The roles, responsibilities, and relations between organizations provide the basis for determining the points of coordination of the processes they execute. In determining the specifics of where and when those processes should intersect, some fundamental questions that govern the management of the processes must be addressed. Because the answers dictate a minimum level of coordination, each organization must be satisfied. Otherwise, the organizations must work together to resolve issues to everyone's satisfaction, that is, to justify a collaborative effort. The paragraphs that follow give these fundamental questions that each organization must satisfactorily answer before program resources are committed to an effort.

B.2.1 What is the question that the investigation is meant to answer?

For the senior-junior relationship, this will probably be at least slightly different for each organization. There should be, however, a logical flow of detail, and the questions of interest to the more junior organizations will support the questions of the more senior organizations. For example:

- Sponsor: What is the best investment strategy for a certain set of constraints?
- EA: What capabilities are required to counter anticipated threat capabilities?
- Executor: What quantitative improvement in the force's ability to perform a specific essential task is achieved through the introduction of a specific capability?

For contemporary organizations, the purpose of their investigations could be considerably different and yet be addressable through a single, well-designed investigation. For example:

- What is the best investment strategy for a certain set of constraints?
- What changes, if any, to current doctrine should be promulgated in response to a change in the threat?

B.2.2 Will the investigation/experiment provide the complete answers desired? What is the sensitivity of the results?

Once an investigative approach is selected, it must be understood to what extent and under what circumstances it will answer the desired questions. A key aspect of this is an understanding of the tools and methods to be employed. Knowledge of the architecture of the experiment, the capabilities and limitations of the tools, and the applicability of the results is essential for generating confidence in the conclusions of the investigation. Without confidence in the conclusions, the value of the investigation is, at best, low.

Whether or not an investigation answers the identified questions is obviously a complex issue. In most cases this cannot be resolved directly; rather, it must be addressed through a series of questions addressing successive levels of detail. The following are examples of some possible questions at the next level of detail:

- What are the tools and methods employed and are they understood?
- What are the conditions or restrictions on the results imposed by the tools, methods, and analysis techniques?
- What assumptions were made in the setup and conduct of the experiment?

B.2.3 What variations are/could be/need to be run?

The answer is as much a function of the tools and methods employed in the investigation as of the central question to be investigated. The basic question of the investigation may require addressing operations in several different types of terrain and thus dictate several variations. The experiment, however, must also be designed with due consideration of the capabilities of the tools. Limitations on the scope of a simulation, for example, may dictate that investigations involving large scenarios be accomplished through the combination of several smaller scenarios. Variations on how those scenarios are combined may then be required. The robustness of the results must also be compared to that required. Increasing the robustness or applicability of the results may also dictate additional variations be investigated. Understanding the number, nature, and extent of necessary or possible variations plays a role in understanding whether the investigation will provide the desired answers, as well as the cost of the investigation.

B.2.4 What is the cost of the investigation?

To remain within resource constraints, it may often be necessary to limit the scope of the desired investigation with respect to the original purpose (questions). It may also become necessary to limit the investigation once the experiment has started, if costs are exceeding expectations.

B.2.5 What is the time required for the investigation?

Just there are cost limits on the investigation, there are usually well-defined limits on when the results of the analysis are required. The same considerations of restricting the experiment because of resource constraints apply to schedule constraints.

B.2.6 Is the execution proceeding as expected?

In some investigations the experiment execution phase may be protracted due to either the complexity of the experiment or the need to conduct several variations. In such instances, it may be beneficial for each organization involved to assess if the experiment is progressing as planned, with the prospect of producing the required information. If it is not, it may be in the best interest of the organization to stop the experiment, end its participation, or insist on modifications. The commitments made to the other organizations obviously play a role in any such decision.

B.2.7 Can the analysis be conducted as planned?

The planned analysis must be designed with some level of robustness. Because experiments most often do not evolve exactly as planned, contingencies must be made for the experiment producing less or somewhat different data than expected. After the experiment is completed, an assessment whether the data produced is sufficient for the analysis as planned must be made. If the data is insufficient, modifications to the assessment will be necessary. In extreme cases, all results of the experiment will have to be discarded and the experiment run again.

B.2.8 An example of coordination within a joint experimentation process

To show how these questions play a role in coordinating the activities of multiple organizations involved in an investigation, consider the processes identified by a program sponsor, EA, and executor as outlined in Figure B-5. Each organization defines the process to execute the tasks for which it is responsible. This can be done without significant coordination between the organizations. Inspecting these processes in the context of the above questions provides insight into the similarities of the processes and where the most intense coordination is required.

B.2.8.1 Developing a new operational vision

This phase occurs before the investigative cycle previously outlined and highlights the unique aspect of joint experimentation. The purpose of this phase is to identify the list of potential operational methods that could seed the discovery process. The methods address a new or revised way of conducting some required operation. They may involve new methods in any or all of the areas of organization, communications, tactics, and improved system capabilities resulting from technology advances.

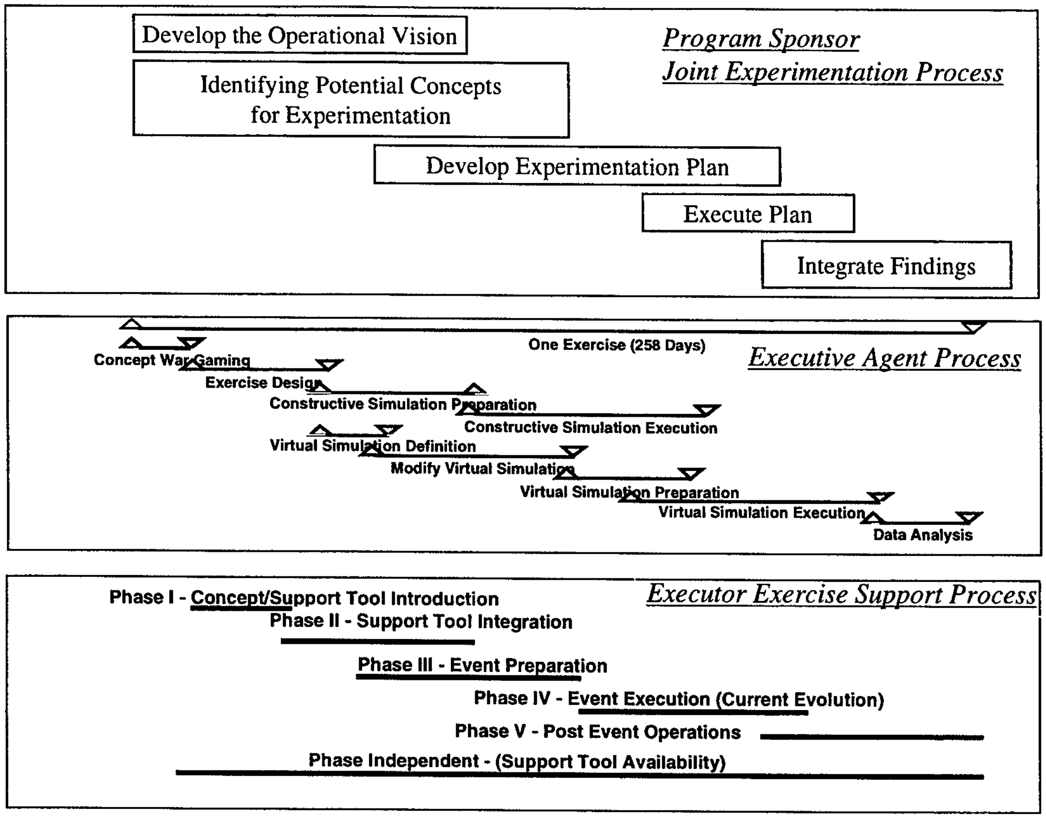

Figure B-5. Example Investigation Processes

In creating this operational vision, it is important to solicit concepts from a broad range of individuals and organizations so that good ideas are not overlooked. Therefore, a relationship between the program sponsor and several organizations in the solicitation and offering of methods contributes to the formulation of the operational vision. This relationship needs to be interactive enough to ensure that the contributed concepts are well understood by those formulating the operational vision. This should primarily be a unidirectional relationship, however, with the program sponsor receiving information and ideas from others.

B.2.8.2 Identify potential concepts for experimentation

The relationship cited above needs to extend into the next box, "Identifying Potential Concepts for Experimentation," because the original contributor of the method involving the concepts probably has ideas on an experimental approach to those concepts and what the limits of such an experiment would be.

Depending on the approach for selecting the organization that will conduct the experiment, another possible relationship may appear in the box, "Identifying Potential Concepts for Experimentation." There are three possibilities for the organization responsible

for conducting the experiment. The organization that provided the concept can be tasked with testing it, the program sponsor can task a trusted agent to conduct the tests, or the program sponsor can conduct the tests. For the first two possibilities, the relationship between the "executive agent" who will conduct the test and the program sponsor will need to develop. In the first case, this development will require an expansion of the relationship between solicitor and provider of concepts as discussed above. In the second case, development will involve an initiation of a new relationship (at least for any specific investigation) between the program sponsor and the trusted agent. For use of a trusted agent, it will also be necessary for the sponsor to establish a relationship with the organizations most knowledgeable with the concepts to be evaluated in the experiment.

The third possibility, that of the program sponsor as executor of the experiment will not be further discussed, except to say that, in general, for this situation, the program sponsor retains the functions of the executive agent. The relationship between the executive agent and the program sponsor would therefore not be required. The other relationships cited as necessary for the executive agent would then become necessary for the program sponsor.

It is during this phase that the formulation of the central questions of the investigation takes place. A decomposition of these questions provides the primary questions of interest to the junior organizations in the experimentation effort. The junior organization must understand the senior organization's question to sufficient extent to ensure that each of its own questions represent a valid subset of the investigation space. The senior organization must understand the junior organization's questions sufficiently to ensure that the sum of the answers to those questions will provide the answers to its own question.

B.2.8.3 Develop experimentation plan

This phase is the period of most intense coordination. It is during this period that most of the fundamental questions cited above must be addressed by each organization. After addressing these questions, each organization must be convinced that continued participation in the investigation is of benefit to them.

What the program sponsor needs from a relationship with the EA during this phase of the investigation is assurance that the plan will address the correct questions and that the results generated will be valid in some sense. The relationship between the program sponsor and the EA must therefore be sufficiently robust to give the program sponsor confidence that the architecture of the experiment provides for the generation and collection of data at the proper level of detail to address the central question. The program sponsor also needs some assurance that the investigation can be conducted within the constraints of the available

resources. It is assumed that the EA will address most of the technical and logistical concerns. The program sponsor's involvement is minimal—he needs only to ensure he is satisfied with the planning. For experiments that depend on specific critical tools or methods, education of the program sponsor on some of their specific capabilities and limitations may also be required.

It is during this phase that the other basic questions common to all investigations (and presented above) must be addressed.

In most cases, the EA will need to rely on the technical knowledge of other organizations to assist in the planning, conduct, and data analysis of the experiment. These other organizations are those with expertise in one or more of the tools or methods to be employed in the experiment. During this phase, what the EA needs from these relationships is sufficient knowledge to ensure that the generated plan is executable and exploits the fullest capabilities of the tools employed. It is also critical that during this phase the EA becomes educated to some extent on the chosen tools and methods. This education will become critical during the execution phase when, as the test director, the EA must provide guidance on the best approach of handling the minor perturbations that are inevitable during experiment executions.

B.2.8.4 Execute plan

In general, the program sponsor should want to know if the experimentation plan was executed without significant deviation. In this case, "significant" would mean anything that would jeopardize the generating or collecting the required data or the validity of the results. For serious issues that arise during the conduct of the experiment, representatives from the program sponsor should be involved in deciding how to proceed. This decision will typically involve real-time modifications to the experimentation plan, which will affect the type, amount, and validity of the data collected. This, in turn, will affect the value of the experiment and the results it produces.

As mentioned above, the EA should act as test director during the execution phase and will provide guidance on handling the minor perturbations to the test plan. It is assumed that the knowledge of the employed tools and methods necessary for this was gained in the previous phase. The critical aspect of the relationship with the experts who will be running those tools is that the experts provide sufficient status reporting so that proper test direction can be conducted. Such information as the nature of any problem, its effect on the execution of the plan, and options for mitigating the effects must be quickly and effectively

communicated. This implies a level of teamwork that must be established and nurtured throughout the event cycle.

B.2.8.5 Integrate findings

If all other steps in the process were properly conducted, this phase should require only very simple and straightforward relationships. The analysis methods should have been previously understood and approved by the cognizant senior organization. Under such circumstances, we are essentially back to a unidirectional information flow. Results of analyses flow up to support analysis of higher level questions. Any flow down is primarily only requests for clarification. The required relationships are therefore primarily simple information provider-requestor types.

Deviations from the simple provider-requestor relationships arise if there were significant deviations from the experiment plan during execution. Under such circumstances, modifications to the analyses will be necessary, and these must again be understood and approved by the proper senior organization. Much communication between all the organizations may also be required to determine whether the original central questions of the investigation can be answered at all, whether additional limitations or restrictions are placed upon the answers, or whether alternative questions can be answered.

B.2.8.6 Summary

The three processes presented in Figure B-5 are now presented again in Figure B-6. This time, however, they are color coded to show periods of concurrency between all the organizations. Regardless of the details of any organization's defined process, that process can be divided into segments that correspond to the five phases shown under the program sponsor in Figure B-6. For the coordination of the specific processes of different organizations, however, there are no "point solutions" where those processes should intersect. Instead, for successful investigations, the critical factor is for the organizations to form an effective team-like working relationship. The "points of intersection" of the processes need to be continuous, with some periods more intense than others. This is not to say that there are not definitely defined areas of responsibility. Nor does it imply any minimum frequency of communication between the organizations. The complexity of the problem and the amount of information and control required to meet each organization's comfort level determine the frequency.

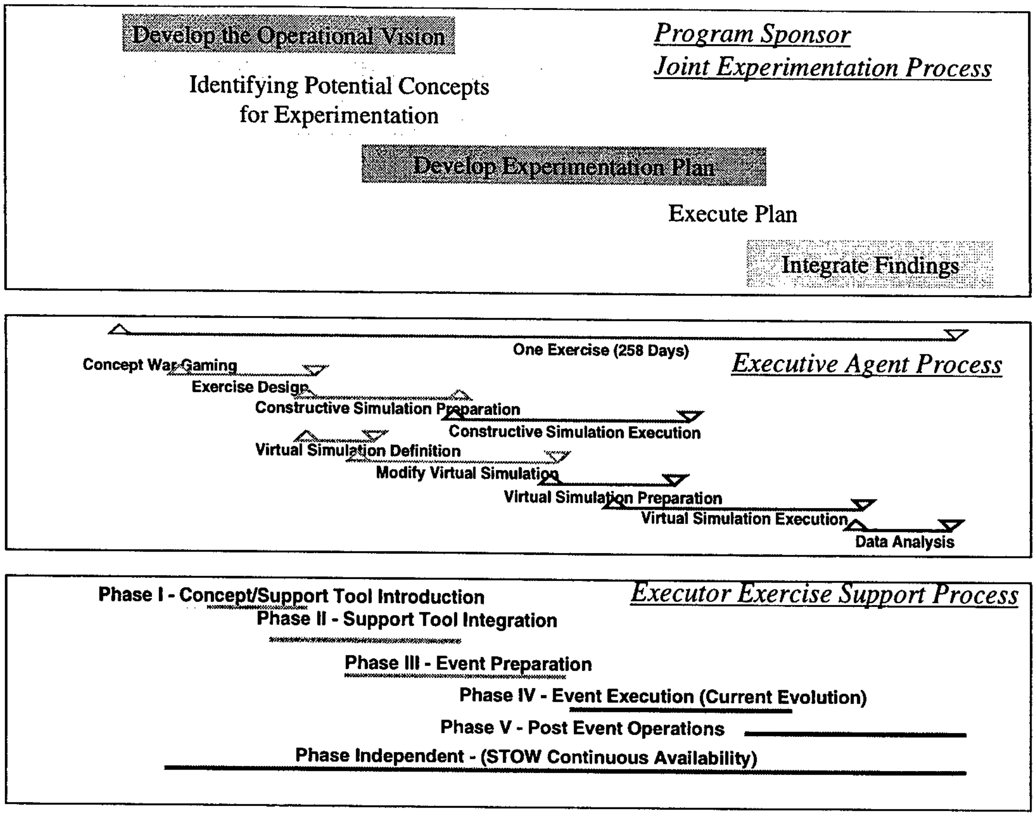

Figure B-6. Example Processes Showing Concurrency

APPENDIX C—MEASURES OF EFFECTIVENESS FOR MODELING AND SIMULATION

APPENDIX C—MEASURES OF EFFECTIVENESS FOR MODELING AND SIMULATION

Based upon the general requirement areas outlined above, measures of effectiveness (MOEs) for any of the methods and tools that support the iterative investigative process can be developed. This was done for the methods of using modeling and simulation in general and the use of the STOW simulation system in particular. The STOW simulation system was then evaluated against these MOEs so that some ideas of the potential and limitations of that system to support analyses associated with solving complex problems could be developed.

The first general requirement discussed above is that of representing the entire problem space. One way of segregating the problem space applicable to military operations is into the battlespace elements, the conditions under which they operate, and their interactions. The elements would include all physical entities of the battlespace. Representation could be as individual entities or as organizational collections such as companies or battalions. The Universal Joint Task List contains a common language of conditions used to describe the operational context in which military task are performed. Conditions are divided into physical conditions such as terrain and meteorological state; military conditions such as mission, intelligence, and sustainment; and civilian conditions such as political policies and culture. Interactions includes two broad categories of physical interactions, such as sensor detections or damage from ordnance, and behavioral interactions, such as proper execution of tactics. Given this division of battlespace representation, the first five MOEs presented in Table C-1 below were developed.

Table C-1. Measures of Effectiveness for Representation of the Problem Space

Requirement Area	Measure of Effectiveness
Represent entire problem space	Can all required elements of the battlespace be represented?
	Can all required conditions of the battlespace be represented?
	Can all required behaviors of the battlespace be represented?
	Do behaviors cover a representative range of competence and other human traits?
	Can all required physical interactions of the battlespace be represented?

To provide consistent results throughout the program cycle, a simulation or family of simulations needs to be able to address investigations at different levels of detail. This requires that the simulation provides an accurate presentation of the situation at any of the needed levels of detail. Reality must not be changed by the selection of the detail of the representation. That is, representations of the situation must be consistent even though resolution differs. In addition to representing the situation at the desired level of detail, the simulation must be capable of providing its output in the format most applicable to the user. This requirement lead to three MOEs listed in Table C-2.

Table C-2. Measures of Effectiveness for Consistent Results at All Levels of Detail

Requirement Area	Measure of Effectiveness
Consistent results at all levels of detail	Does the simulation provide the required data at the right level of detail (resolution)?
	Are investigations at multiple levels of detail supported?
	Are the output formats of the simulation flexible enough to support all user requirements?

Simulations are simply someone's concept of a version of reality. Not all aspects of any situation can be completely simulated. Therefore, by the very nature of simulations, some aspects of reality are made abstract in the representation. The developer of the simulation must decide which aspects are critical to the representation and which can be abstracted away. The user of the simulation must therefore be aware of which aspects of reality are represented by the simulation and which are abstracted away. He must then decide if a simulation that abstracts specific aspects of reality meets his investigative requirements.

The next step is to determine if the simulation provides a faithful representation of those aspects it purports to represent. The process of describing how a simulation represents reality and how well it does it are typically associated with verification and validation. Although the formal procedures of verification and validation support the development of user confidence, such procedures are not absolutely necessary to establish that confidence, nor are they entirely sufficient. Three MOEs are included in Table C-3 to capture the essentials aspects associated with establishing user confidence in the representation the simulation presents. Two additional MOEs are provided to address the user confidence in estimating the costs associated with using the simulation.

Table C-3. Measures of Effectiveness for User Confidence

Requirement Area	Measure of Effectiveness
User confidence	Are the abstractions of reality necessary in simulations clearly documented?
	Are the effects of the abstractions of the simulation on any supported analysis defined and acceptable?
	Does the simulation faithfully represent what it purports to represent?
	Are the cost and schedule estimates and variability for employment of the simulation understood?
	Are the confidence limits on cost, schedule, and performance variability understood and within acceptable limits?

Many simulations have stochastic aspects. Because of this, multiple simulation runs starting from the same initial conditions should be expected to evolve slightly differently. The significant results at the level of detail under investigation, however, should be consistent from run to run or between trials consisting of several runs. The statistics generated on the probability of success for a given tactic, for example, should be consistent within some allowable variance between trials. Without such consistency, it is difficult if not impossible for the analysis that the simulation supports to be meaningful. The requirement for repeatability has led to two MOEs presented in Table C-4.

Table C-4. Measures of Effectiveness for Repeatable Results

Requirement Area	Measure of Effectiveness
Repeatable results	Is the generated data reliable and repeatable?
	What is the variance of the results from multiple simulation runs and does this affect any supported analysis?

The quality of adaptability has several aspects. During the course of the iterative problem-solving process, some investigations will be of relatively major scale and some will be very minor. Ideally, the same simulation tool should be capable of supporting all scales of investigations. Also, at some points in the process, a quick, inexact answer is required, but at other times, more precision is required. This fidelity of the information produced is distinctly different from the level of detail or resolution discussed earlier. Investigations can address different levels of fidelity at the same level of resolution, for example. Consistency of results across investigations of varying levels of fidelity is desired. Such consistency may well be easier to ensure with one simulation if it could provide results of varying fidelity.

When used in an investigation, a simulation must be capable of representing the desired conditions and situations under which the object of the investigation is to be tested. This is partially addressed by the extent to which the battlespace can be represented; however, situations during the execution of an investigation often unfold in unforeseen or unpredictable ways. It is important, therefore, that the investigator have the ability to control the situation to ensure that it remains as close as possible to the desired conditions of the test. It thus becomes important that the investigators have the ability to control the evolution of the simulation. It is also important that they have the flexibility to control aspects of the review of the data generated during the execution. This need for control led to the MOEs associated with adaptability listed in Table C-5.

Investigations may often employ the use of multiple methods or tools simultaneously. Simulations may be used to supplement or augment live tests or expert seminars. Depending on the complexity of the investigation, multiple simulations may be required. It is therefore important that these methods be capable of operating together in some manner. For a seminar that simply uses results from a simulation to assist experts' deliberations, the previously discussed areas addressing suitability of outputs apply. For live exercises, use of virtual trainers, and expert seminars using continuous interaction with simulations through systems other than the simulation's normal operator interfaces, the ability of the simulation to interoperate with these systems becomes important. Table C-5 presents the MOEs associated with the above aspects of adaptability.

Table C-5. Measures of Effectiveness for Adaptability

Requirement Area	Measure of Effectiveness
Adaptability	What are the limits of scalability of the simulation?
	Is the simulation capable of being interfaced to any live or virtual system?
	Is the simulation capable of being interfaced to other constructive simulations?

One of the major advantages offered by simulation is the potential for the control of specific aspects of the investigative event. Ideally, a simulation should allow control of the location, structure, strength, and capabilities of represented forces. It is also desirable to have control over the environmental, military, and civilian conditions under which the simulated operations take place. Control of such aspects should be available during both initialization and execution of the simulation. Simulations also provide the potential for extensive and flexible collection of data generated during the event. It is important, therefore, that a

simulation system offer extensive capabilities to present this data in the format most appropriate for the analyses to be conducted. Table C-6 presents the MOEs associated with simulation control.

Table C-6. Measures of Effectiveness for Simulation Control

Requirement Area	Measure of Effectiveness
Execution/Review Control	What are the capabilities for controlling force location and materiel condition?
	What are the capabilities for controlling force behaviors?
	What are the capabilities for controlling apparent event time (including pausing, speed up, slow down, jump ahead or back in time)?
	What are the capabilities for controlling other conditions (environmental, political, cultural)?
	What are the capabilities for reviewing data (type and format of data, control of apparent time)?

For a method to be of value to an investigation, it must be capable of providing its results within the timeframe of the investigation. That is, all the resources required to execute the method must be available within that timeframe, and the time to execute the method cannot be excessive. Three of the measures listed in Table C-7 reflect these concepts.

Table C-7. Measures of Effectiveness for Schedule

Requirement Area	Measure of Effectiveness
Schedule	What range of time is required to modify the simulation to address the specific issues of the investigation?
	What range of time is required to initialize the simulation for specific experiments?
	Can the desired quantity of data be generated in the time frames allowed?

Another major requirement area is cost. The measures listed Table C-8 indicate that it is closely related to all the other requirement areas discussed. The costs to maintain, adapt, and operate the simulation are dependent on the simulation's performance in the areas of the other MOEs. This correlation is obvious from the cost-related MOEs derived for this study. For software-intensive systems like simulations, most of the associated maintenance and operation costs are the work hours required to perform them.

Table C-8. Measures of Effectiveness for Simulation Costs

Requirement Area	Measure of Effectiveness
Cost	What is the level of effort required to add new battlespace elements?
	What is the level of effort required to add new battlespace conditions?
	What is the level of effort required to add new battlespace behaviors?
	How easy is it to change programmed tactics and behaviors?
	What level of effort is required to accommodate interactions with new entities?
	What level of effort is required to add new interfaces to live or virtual systems?
	What level of effort is required to interface to other constructive simulations?
	What level of effort is required to generate scenarios and instantiate units?
	What level of effort is required for simulation control?
	What level of effort is required to plan and prepare data collection and extract collected data?
	What level of effort is required to prepare for new events?
	What level of effort is required to execute events?
	Is significant education on the system required to obtain satisfactory results?
	What effort is required to maintain the system?
	What level of effort is required to maintain the software?
	Are any special facilities or expertise required for life-cycle maintenance?

The final requirement areas that a simulation must address are those of reliability and availability. The simulation must possess the operational stability to provide complete executions of the required representation within the allotted time and budget. Table C-9 lists the MOEs associated with reliability and availability.

Table C-9. Measures of Effectiveness for Reliability and Availability

Requirement Area	Measure of Effectiveness
Reliability	Does the simulation's reliability support timely and cost-effective generation of the required data?
Availability	Does the availability of the system support timely and cost-effective generation of the required data?

APPENDIX D—EVALUATION OF THE STOW SIMULATION SYSTEM

APPENDIX D—EVALUATION OF THE STOW SIMULATION SYSTEM

The STOW simulation system was evaluated against the utility MOEs outlined above to provide a measure of STOW's ability to support the complex problem-solving process.

D.1 REPRESENT ENTIRE PROBLEM SPACE

Can all required elements of the battlespace be represented?

The STOW simulation system can represent any physical battlefield entity, condition, phenomena, or effect, as demonstrated by the inclusion of most battlespace entities in the current STOW simulation system. The STOW simulation system includes representations of wheeled and tracked vehicles, aircraft, and sea surface and subsurface vessels. The sensors and weapon systems associated with these platforms are also represented. A limited set of cultural features and facilities that may be of tactical or strategic importance is also currently included in STOW.

There are some entity classes or feature types that are not currently represented in the STOW simulation system. For example, there are no satellite representations within the STOW simulation system. Absent also are functional relationships such as facility operation linked to the state of power grids. Although these features are not currently in the STOW simulation system, the capability of the system to accommodate such representations is clear. The ease or difficulty of adding any desired battlespace element not in the current system is discussed later in this evaluation.

Representations in the STOW simulations are at the platform, entity, system, or feature level, such as tanks, planes, ships, radars, missiles, buildings, and bridges. The software architecture upon which the STOW simulations are built can accommodate models of either higher or lower resolution. Although modeling at different levels of resolution is not an issue, interactions between models that represent the world at different levels of detail is problematic. Extreme care and extensive effort would be required to ensure that each cross-resolution interaction was valid. Therefore, although it is possible to represent all elements of the battlespace in the STOW simulation, a practical limitation is that representation only at the entity and major system level of resolution is easily accommodated.

Can all required conditions of the battlespace be represented?

The Universal Joint Task List (UJTL) defines conditions as those variables of an operational environment or situation in which a unit, system, or individual is expected to operate that may affect performance. It lists the three categories of conditions as physical, military, and civil. The STOW simulation system has extensive representations of the physical conditions of terrain, air, sea, energy propagation, and gravity. As there are no satellites in STOW, representations of space conditions have not been developed to date. The architectures for the representation of physical conditions are, however, state of the art. They are easily capable of not only representing any physical condition to the resolution required, but these conditions can be accurately represented as dynamic or changeable over time.

Military conditions that relate to materiel state, movement, firepower, and force level are easily accommodated within the STOW simulation system. Conditions that deal with personnel capabilities such as nutrition and health, physical conditioning, morale, and fatigue can be accommodated within the STOW simulation system, but currently are not represented. Representation of such conditions would require considerable knowledge acquisition before they could be confidently represented in any simulation. Representation of military conditions that deal with interpersonal relations, such as staff or multinational integration, would require realistic models of the effects of the various possible states of such relations. Although such models are conceivable, significant development would be required to produce them. The STOW architecture could accommodate such models if they were at the appropriate level of resolution, but because they cannot realistically be expected in the near future, it should be considered that any simulation system, including STOW, cannot represent this class of military conditions.

Representation of civil conditions such as political policy, culture, and economy is in a state analogous to the interpersonal military conditions discussed above. Modeling of the effects of these conditions is difficult at best, and it is doubtful that any simulation will be capable of reliably representing these factors in the near future.

Can all required behaviors of the battlespace be represented?

Behaviors are represented in Joint Semi-Automated Forces (JSAF) through two different means. Basic behaviors of entities and lower echelon units are represented as finite-state, discrete, well-defined tasks. More complex logic, especially for higher echelon units, can be incorporated using more sophisticated software representations such as artificial intelligence. The ability to accommodate different behavioral representations means that the

most appropriate method can be selected. Appropriateness is user defined based upon requirements in areas such as computer resources, speed, and fidelity.

The STOW simulation system currently has a significant number of behaviors represented in task frames. Representation in task frames is computationally efficient for simple behaviors, but becomes extremely difficult as the complexity of the behaviors increases. This places a practical limit on the suitability of task frame modeling of behaviors. For Army units, task frame representation of behaviors at the company level becomes extremely difficult. Behaviors for units above that level are simply too difficult to generate in the discrete finite state task representation. For higher level unit tasks, more sophisticated software constructs such as artificial intelligence or constraint satisfaction must be employed. The STOW simulation system has demonstrated its ability to embody these more advanced software representations. It currently includes some selected behavioral representations for entities and unit commanders using such advanced representations.

Considerations on the level of effort required to add or modify behaviors in the STOW simulation system are discussed later in this evaluation.

Do behaviors cover a representative range of competence and other human traits?

In general, variances in the simulated combatant to account for factors such as expertise, aggressiveness, courage, or intelligence are not contained in the STOW simulation system. The architecture allows for such variance, but the models to support it do not yet exist. How each of the above human factors manifests itself in behavioral effects is an area of active research. Therefore, such variances are currently beyond the ability of simulations to reliably represent them.

Can all required physical interactions of the battlespace be represented?

Movement of entities in STOW is related to the environment in which it is operating. Surface type and grade affect vehicles on land. Vessels in the water respond to sea state and currents. Wind direction and velocity can affect aircraft speed over the ground, but this specific aspect is not currently modeled.

All basic battlespace interactions such as detection by any means, engagements, and communications are accommodated within the STOW simulation system. Sensing is done through physics-based sensor and propagation models. As such, sensor classes using any form of energy can be represented. Because the models are physics-based, the effects of propagation and countermeasures can be included in sensor modeling.

Weapons effects are accommodated through the categorization of munitions into classes and the use of damage-assessment tables specific to each entity. The system has the capacity to make these tables as complex as necessary. In this manner, the effects of any weapon on any entity can be represented.

Communications within the synthetic forces is via Command and Control Simulation Interface Language (CCSIL) messages. These messages pass information representative of what would be actually passed between units and commanders. Representation of communication content is thus present in the STOW simulation. This capability allows for the modeling of some information warfare capabilities such as message intercept, deception, and transmission errors. STOW currently does not provide for the physical modeling of communications transmissions, so it does not directly accommodate transmissions losses or the possibility of jamming. Models for communications networks' loading and delays are also not currently part of the STOW simulation. Again, these effects could be accommodated within the STOW simulation with the proper lead time and application of resources.

D.2 CONSISTENT RESULTS AT ALL LEVELS OF DETAIL

Does the simulation provide the required data at the right level of detail (resolution)?

The STOW simulation does not directly provide representations at multiple levels of resolution, but only at the entity level or finest level of detail available. There is no way to "select" the level of detail of the data that the simulation provides. It is not, however, the only level of data that can be obtained; aggregation of detail, particularly such aggregations as results of force on force engagements, can also be obtained from the system.

The level of resolution provided by the force simulation within STOW is at the combat entity level. Position, movement, engagements, and damage at the combat entity level are provided. The STOW simulation supports military utility assessments that require such data.

Higher level assessments can be supported through repeated simulations with changes in specific conditions. For example, the effectiveness of tactics can be investigated through multiple simulations with the synthetic forces employing various tactics. In addition, the effectiveness of the tactics can be evaluated under a variety of differing conditions such as weather, terrain, and opposing force structure. Thus, judicious use of the STOW simulation system can provide data with any level of detail down to entity-level engagements and status.

Are investigations at multiple levels of detail supported?

STOW simulates the battlespace at the level of individual entities such as tanks, planes, ships, and missiles. The state, emissions, and performance of individual sensors are modeled. The logistics state of the entities with respect to fuel and ammunition is modeled. Communications are modeled to the level that proper communications equipment must be present and operational. The content of messages can be passed between entities and is available for interfacing with real-world command, control, communications, computers, and intelligence (C4I) equipment. Investigations to this entity level of detail are therefore possible with STOW. Through proper aggregation and consolidation of the events and conditions of the represented battlespace, investigations at any level of detail, up to theater, are possible. Such higher level investigations are contingent upon the investigation not being critically dependent upon some aspect of the battlespace that is abstracted away from or not represented in the STOW simulation.

Are the output formats of the simulation flexible enough to support all user requirements?

The standard output formats of the STOW simulation system during run time are two-dimensional plan view displays and three-dimensional stealth displays. The plan view displays provide a dynamic map view of the battlespace. Units and features are represented by icons, and the unit icons move as the units do. Interactions are displayed, and the icons change for destroyed units. The scale and center point of the map is selectable. The three-dimensional stealth display provides a view that immerses the observer in the battlespace. This view often provides insight into behaviors and interactions that is not evident from the plan view display. It is therefore helpful in understanding why events transpired as they did. That visualization products of the type provided by the STOW simulation system help decision makers absorb and comprehend information seems intuitive, but has yet to be rigorously proven.

If the STOW simulation system is run in conjunction with C4I systems, the messages send by STOW to those systems is another form of output. Because the type of report, frequency, and content can be tailored, this form of output provides great flexibility to the STOW output options. The content of these messages is, of course, limited by the content of the data within the simulation. For example, messages detailing the type of damage sustained by a ship's electrical generator cannot be generated if STOW does not model damage to that resolution.

All entity state and interaction data that is transmitted across the Run Time Infrastructure (RTI) is captured by a data logger for possible playback and review after the event. This logger data is also played into the After Action Review System (AARS) for further sorting and analysis. The AARS uses programmed data-collection agents to filter desired data and store it in spreadsheet form. Once this data is in Excel spreadsheets, the analyst has a variety of options on how to display it within any program of the Microsoft Office suite. The data can also be displayed in any other program that understands the Object Linking and Embedding (OLE) protocol or that can import Excel files or accept one of Excel's export formats.

D.3 USER CONFIDENCE

Are the abstractions of reality necessary in simulations clearly documented?

The aspects of what is simulated and how it is simulated are clearly documented in STOW. Specific abstractions are not documented. The user can, however, easily infer if any specific abstraction was made in the construction of the simulation through the documentation that describes the simulation. This is the practical approach, since it is easier to list what is in a simulation that it is to list what is not present in a simulation. In addition, any list of what is not in a simulation is certain to be incomplete with respect to the specific requirements of all potential users, many of whom may not even be known at the time of the simulation documentation.

Are the effects of the abstractions of the simulation on any supported analysis defined and acceptable?

During its development phase, the design purpose of the STOW simulation system was to support training. No attempt was therefore undertaken to determine or document the limitations of the STOW simulation system with respect to any limitations on the applicability to analysis.

Does the simulation faithfully represent what it purports to represent?

Many of the individual algorithms used in STOW have undergone that process. In addition, every subject matter expert evaluation of the system performance with respect to any of the applications for which STOW has been used to date resulted in a finding that the simulation does represent what it purports to represent.

Exhaustive testing of all combinations and permutations of conditions possible within the simulation and presented to the simulated forces for their reaction is difficult to

accomplish. Schemes to measure and describe the probability of faithful or realistic representation under any given set of conditions need to be developed before a useful characterization of the STOW simulation system could be made.

Are the cost and schedule estimates and variability for employment of the simulation understood?

Because of the complexity of the system and the uniqueness of the requirements for each event, creation of an algorithm that could provide time and cost estimates for the preparation and execution of an event is not possible. Development of these estimates is, therefore, dependent on the knowledge and experience of the experts on the system. These experts have extensive experience in the development and modification of simulations, the interfacing of simulations to live, virtual, and other constructive systems, and the execution of simulation events. The factors to be considered in the development of cost and schedule estimates for the employment of the STOW simulation system are thus well understood by these experts.

Are the confidence limits on cost, schedule, and performance variability understood and within acceptable limits?

Because of the experience and expertise of the experts producing the cost and schedule estimates, there is usually high confidence of their accuracy. If situations should arise where the experts feel less certain about their estimates, they are still experienced in placing a confidence limit on their estimates and will provide accurate but greater bounds on their possible variability.

D.4 REPEATABLE RESULTS

Is the generated data reliable and repeatable?

The STOW simulation contains stochastic aspects; the nature of data transmission in distributed simulations is one contributor. Therefore, many of the details of the manner in which a specific scenario unfolds are not repeatable. That is, given the same initial conditions, no two runs of a scenario should be expected to be exactly the same. The general nature of the simulated conflict, however, is consistent from run to run. That is, the results are generally reliable and repeatable as long as the detail sought is not beyond the realistic capability of the system. For example, if the investigation is interested in expected improvements in force survivability through employment of a new tactic, results of individual runs will have some variability, but will be generally consistent overall. If the investigation is interested in the determining the length of time into a battle that a specific

tank will survive under employment of the new tactic, results could conceivably vary greatly between runs.

What is the variance of the results from multiple simulation runs and does this affect any supported analysis?

Just as a simulation could be used to answer questions at different levels (what were the number of mines neutralized per unit time? were operations more effective with system X or system Y? did we win the war?), one could address the variance relative to these different levels. It is likely that variance would decrease as the level detail of the question decreased.

For the most detailed questions, there will not be sufficient data to make a solid statistical case about expected standard error. Even if there were, the number of variables associated with any scenario is too great to argue that the results have general applicability. The methods we would need to say something statistically sound would take a number of months to develop.

D.5 ADAPTABILITY

What are the limits of scalability of the simulation?

The limits to which the STOW simulation can be scaled have never been absolutely determined. Neither all the factors that affect the limits of scalability nor the effects of those factors have been defined. The number of factors that affects the scalability of distributed simulations makes the definition of the limits of scalability very conditionally dependent. Choices the designer of an application of the simulation system makes determine the size of the simulation that can be run. Providing meaningful statements on the scalability of a distributed simulation requires technology advancements in metrics and the tools to measure system performance with respect to those metrics.

Given all the dependencies that scalability has and the limits of current technology in the areas of metrics and tools, the best that can be provided is a statement of the size of simulations that STOW has achieved in the past. During the STOW 97 event, a scenario was executed that contained more than 8,000 active entities.

Is the simulation capable of being interfaced to any live or virtual system?

The interface between a constructive simulation and live forces is via the C4I systems used by those forces. The STOW has successfully interfaced to a representative sample of such systems. These interfaces are essentially one way, in that the state of the simulated battlespace can be output to the C4I systems, but direct instantiation, ordering, or control of

the synthetic forces through C4I systems is not currently possible without a person in the loop. There is potential for total integration of the STOW simulation system with any given C4I system because STOW uses the CCSIL message concept to pass information between synthetic units.

Interfaces to virtual systems are a hybrid of the interfaces required for live forces and other constructive simulations. If interfaces between the operator of the virtual trainer and the simulation are required, they will be via the operator's normal C4I systems. Interactions between virtual and synthetic forces must be handled via protocols similar or identical to the protocols used between simulations. All the capabilities, potential, and issues involved in the previously discussed interfaces must therefore be considered when interfacing to virtual systems.

Is the simulation capable of being interfaced to other constructive simulations?

To address interoperability between constructive simulations requires addressing two primary concerns: the simulation timing control and the data exchange interface. Timing for simulations falls into two major methods, scaled real time and event-based stepping. The simulations within STOW are based on real time. They have little flexibility in their run-time speed, so interoperability of the STOW simulations with other constructive simulations essentially requires that those simulations be capable of running at real time. Interoperability with simulations that run at something other than real time is theoretically possible, but a means to keep the simulations synchronized would have to be developed.

The second area of concern in constructive simulation interoperability is definition of the data exchange interface. Simulations that comply with the DoD High Level Architecture (HLA) address their data exchange interface through the definition of a Federation Object Model (FOM). The STOW simulation system has defined a FOM derived from the standard Distributed Interactive Simulation (DIS) protocol, but expanded to meet specific needs of the system. Other simulations will define a FOM for their purpose. In the absence of development of DoD-wide standards, it should be expected that several different FOMs will be developed, although occasionally two or more simulations may agree on the use of a common FOM.

For interoperability between the STOW simulation system and a constructive simulation using a different FOM, some initial interface definition effort will be required. Once the data to be exchanged between simulations are identified, a method of resolving differences between the FOMs must be chosen. There are basically two options for this. First, one FOM—either an adoption of one of the existing FOMs, or some combination—can be

agreed upon. Doing so will most likely require modification to one or both simulations. Second, essentially a translator can be used between FOMs. For this method, neither simulation will likely require changes, but the development of a FOM-to-FOM mapping is required.

Work in the second method of addressing the data exchange interface mentioned above is ongoing. The incorporation of an "agile FOM" translation layer is meant, within limits, to facilitate this method and provide a structure for the definition of the translator. The initial product of the agile FOM development effort will be a translation between the STOW FOM and the Real-time, Platform Reference Federation Object Model (RPR FOM), which is also based upon the DIS protocol. Interoperability with any simulation that uses either the STOW FOM or the RPR FOM should be relatively simple from a data exchange perspective. Interoperability with other simulations will be more complex, with the level of complexity proportional to the differences in the FOMs. A gross indication of complexity can be obtained by comparing the levels of resolution at which the simulations operate. The greater the difference in resolution level, the more difficult achieving interoperability will become.

D.6 EXECUTION AND REVIEW CONTROL

What are the capabilities for controlling force location and materiel condition?

Force location is controllable on an entity-by-entity or small unit basis. Units can be placed anywhere in the battlespace that makes physical sense (i.e., ground units cannot be placed in the air, etc.). In addition, entities or units can be moved nearly instantaneously to any other location in the battlespace.

The materiel status of forces represented in the simulation is controllable. Fuel and munitions states can be set at any desired level. "Magic repair" of damaged or destroyed entities is not allowed per se; however, damaged or destroyed entities can be deleted from the simulation and fully functional new entities with the same organizational information such as associated units and call signs, can be added in place of the old entities. In addition, any entity can be added to or deleted from the battlespace at any time. This capability must be used with care, however, as such changes are reflected in the collected data. If not explicitly accounted for in post-event processing, adding or deleting units could corrupt the after action review or analysis.

What are the capabilities for controlling force behaviors?

Total control of an entity's behavior is possible. Entities can be tasked as desired, and the operator can change their tasking at any time. An operator also has explicit control over

whether a sensor is on or off. In addition, an operator can take manual control of an entity and control its movement, firings, or any other actions accommodated in the simulation. Because the operators typically are assembled into teams that control the friendly or opposing forces, coordination of operator actions allows the behaviors of the overall force structures can be controlled at a more strategic level.

What are the capabilities for controlling apparent event time (including pausing, speed up, slow down, jump ahead or back in time)?

The STOW time management scheme is scaled real time. The STOW simulation system is capable of operations at several times real time. For simulations that can be conducted on a single machine, ratios as high as 100:1 can be achieved. Although the STOW simulation is capable of running faster than real time, it is not normally executed at anything but real-time speed. Some anomalous behaviors are possible when executing at speeds faster than real-time, and the effects on the validity of such a simulation are highly dependent on the scenario that is run, the resources applied, and the scale factor of real time at which the simulation is run. For fixed computational resources, running faster than real time generally requires reducing the number of entities being simulated. An additional consideration is the most typical applications of the STOW simulation system require a human-in-the-loop (HITL). HITL support makes faster than real time operation infeasible because human performance seriously degrades when inputs occur at a faster than real-time rate.

The STOW simulation system is designed for continuous real-time execution of a scenario, in great part because the original intended training applications required a human-in-the-loop. The STOW simulation system is not capable of speeding up and slowing down as desired during scenario execution. The STOW simulation system is capable of pausing, and pausing is routinely used for saving scenario state during execution. Such state saves, or "checkpointing," provide the capability to return to a previous point in the scenario and move forward again from that point. Because of the stochastic nature of the simulation, a second execution from a saved checkpoint is likely to differ from the first execution is some manner. STOW is not designed to jump ahead in a scenario.

The capability to jump ahead in time during scenario execution presents two challenges. The technical challenge is how to do this and ensure that the jump produced an operationally plausible evolution. That is, the results of the jump must be valid and represent a possible portrayal of reality. The second issue is that for application with an HITL, care must be taken that such forward jumps in time are not presented in such a manner as to cause the operator to become disoriented or confused. Because of the complexities of both of these

issues, the STOW simulation system did not attempt to address the capability of jumping ahead in time during scenario execution. Using the STOW simulation system, it is possible to create separate scenario files that represent a plausible time line evolution of a specific military operation and load and execute them in succession. The effect of jumping ahead in a scenario is thus possible, but requires additional effort on the part of the event or experiment designer and care in application if there are humans-in-the-loop.

What are the capabilities for controlling other conditions (environmental, political, cultural)?

The STOW simulation system provides a comprehensive representation of the physical battlespace. Any section of the world can be accurately depicted, or artificial terrain can be created. The effects of the environment and operations effects on the battlespace can be simulated using three major technologies developed for the STOW program.

The Total Atmosphere Ocean Services (TAOS) Environmental Service System is a system for collecting live and historical environmental data from a variety of external sources; merging these data into a uniform, "seamless" representation of the environmental state; and distributing this state to the simulation applications participating in a distributed simulation event either in real-time or by means of predistribution. The representation of the environmental state is in the form of a gridded atmosphere ocean and of surf data that vary in three spatial dimensions and in the time dimension.

The Dynamic Virtual Worlds (DVW) technology allows simulations to include a range of real-world local and ambient environmental effects. Examples of local effects include dust generated by vehicle movement or ground-level munitions detonations, smoke from fires, and signal and illumination flares. Examples of ambient effects include clouds, wind, precipitation, ocean waves, variations of illumination with time of day, and hydrographic effects on terrain. In addition to providing the environmental representation of these effects, the DVW architecture includes provisions for modeling consequence of these effects on the force simulations.

The Dynamic Terrain and Objects (DTO) technology generates and communicates geometric and feature changes to the terrain database. DTO modifies this data base in real time, affecting the critical aspects of ground interactions including intervisibility, trafficability, damage to bridges and buildings, and other forms of cover and concealment. In addition to run-time alterations, the DTO technology allows pre-event tailoring of the synthetic natural environment through addition or modification of buildings and other structures and emplacement of combat engineering works.

The STOW simulation system does not explicitly model other conditions such as political and cultural factors.

What are the capabilities for reviewing data (type and format of data, control of apparent time)?

Any information transported via the RTI can be captured for later review. This includes messages from C4I systems that have been interfaced with the STOW federation and human-in-the-loop decisions manifest in information exchange in the simulation network. Data collection can be tailored from "all" to only desired specific pieces. Data is collected in two formats: as logged from the RTI and as filtered and processed to support specific after-action data requirements. The data to be filtered and processed, along with the algorithms for these operations, can be as specified by the end user. The collected data can be flagged for inclusion in a new or modified report, formatted as desired by the end user. In addition to the statistical analysis of the data that can be included in and formatted reports generated by the AARS, data can be exported from the AARS to spreadsheet and scientific programs in any specified format. Also, the architecture of the AARS allows for comparison of data across multiple simulation runs.

The AARS is capable of relating event data to measures of performance (MOPs) associated with specific tasks in the UJTL of service-specific task lists. This capability provides the options to substitute specific test plan objectives for UJTL tasks to provide the linkage between collected and processed data and the objectives of the test plan. The user can also define thresholds for effectiveness or objectives for generation of red/yellow/green type displays for quick read on the level of success. The AARS is also capable of interfacing with a "personal digital assistant" to modify (in real time, if required) the previously entered standards associated with specific MOPs or objectives.

Playback of the simulation can be performed with either two- or three-dimensional views.

Data files can be indexed for replay at faster than real time, but replay is then limited to the speed determined by that indexing. Workstations can be set up on site, connected to the event network, and the information it contains can be viewed at any location through web browser technology.

The AARS is not a real-time system; there is always a delay in access to simulation information. Data captured by the logger is sent to the AARS for its processing at intervals determined by the operator or required to support the experiment. For practical reasons, these intervals should be on the order of tens of minutes.

In addition to the capabilities of the AARS, STOW allows for any screen from any workstation can be captured and saved or sent to a printer. In addition, specific displays can and have been constructed to provide up-to-the-second historical and status data on specific entities or units.

D.7 SCHEDULE

What range of time is required to modify the simulation to address the specific issues of the investigation?

The time required to tailor the STOW simulation system to provide the precise information required for an investigation is dependent upon the nature and degree of the tailoring required. The steps that must be considered for any modification include knowledge acquisition and engineering to ensure that the modification is well defined and the resulting model will be valid, coding of the changes, and testing to ensure that the simulation operates properly with the incorporated modifications. For minor modifications to entity capabilities or behaviors, these steps can be accomplished on the order of one to a few weeks per modification. For major modifications, such as a required change to the system architecture, several months may be required. The development team associated with the STOW program is experienced with incorporating changes and can provide accurate lead time estimates for any desired change.

For large or complex events, it is also probable that new requirements for modifications to the simulation software will be identified during the exercise preparation phase. For the identified changes, all aspects or implications of the required changes will not be known when the need for the change is first identified. These two factors contribute to the probable need for some software modification iterations. That is, a modification will be made, tested, and further necessary modifications identified. This process will be repeated until an acceptable version of the software is developed. As an example of the number of different software "builds" that may be necessary to support an event, Table D-1 lists the software build versions (and release dates) that were necessary for the support for JFCOM's JE9901 event.

Table D-1. JSAF Builds to Support JE 9901

JSAF Version	Release Date	Event
<4.0		
4	03/03/99	Integration Event 1
4.1	03/25/99	Fed Integration Event
4.2	04/15/99	Integration Event 2
4.2A	04/19/99	
4.2B	04/20/99	
4.2C	04/21/99	
4.2D	04/22/99	
4.3	04/29/99	Rehearsal 1
4.3A	05/02/99	
4.3B	05/03/99	
4.3C	05/05/99	
4.4	05/12/99	Rehearsal 2
4.4A	05/14/99	
4.4B	05/17/99	
4.4C	05/18/99	
4.4D	05/19/99	
4.4E	05/20/99	
4.5	05/27/99	
4.5A	05/28/99	
4.5B	06/01/99	
4.5C	06/02/99	
4.5D	06/03/99	
4.5E	06/07/99	
4.5F	06/07/99	Trial A Start
4.5FA	06/08/99	During Trial A
4.5FB	06/08/99	During Trial A
4.6	06/23/99	Prior to Trial C
4.6A	06/28/99	Prior to Trial C
4.6B	07/07/99	Trial C Start
4.6C	07/12/99	Trial D Start
4.7	07/24/99	
4.7A	07/26/99	
4.7C	07/27/99	
4.7D	07/28/99	
4.7E	07/29/99	
4.7F	07/31/99	Trial E Start

continued

Table 4.1 (continued)

JSAF Version	Release Date	Event
4.7FA	08/06/99	
4.7G	08/19/99	Trial G Start
4.7GA	08/25/99	During Trial G

What range of time is required to initialize the simulation for specific experiments?

Initialization time can vary with the complexity of the scenario to be run. Extensive scenarios can require 2 or more weeks for the creation of the scenario files used to initialize the simulation. Small scenarios can be created in less than an hour.

Can the desired quantity of data be generated in the time frames allowed?

The STOW simulation system runs a real-time simulation. The data it generates on entity level interactions cannot, therefore, be gathered faster than real time. Scenarios that would take several hours to evolve with live forces will take several hours to simulate with the STOW simulation system. Some time compression can be obtained, however, by running scenarios with different conditions in parallel, should processing and staffing resources allow for this.

D.8 COST

What is the level of effort required to add new battlespace elements?

The effort required to add new battlespace elements involves code development and knowledge acquisition (KA), the two main efforts necessary to quantify the vital parameters associated with the element. The two are not completely independent, however. The KA effort must deliver a product in a form usable to the software developers. The software developers must work with the KA agent to ensure understanding of the modeling method so that the proper KA product can be delivered. Further, the software developers must test their product against the KA description and work with the KA agent to design acceptable tests. Such coordination efforts must be accounted for in the level of effort estimates for both KA and software development.

The simplest battlespace elements to add are entities. The force simulations used in the STOW simulation system uses a common model approach to the physical representation of battlespace entities. To add new entities, whether simple modifications of existing entities or those that can be created from existing common models, requires little programmer effort. The level of effort required to add such entities depends on the complexity of the element.

Simple entities with few systems or components can be coded and tested in as little as a staff week, with most of that time required for software testing. More complex entities can take up to a few staff months of effort. For entities that are unlike any currently represented or for which the current common models are not adequate, code development and testing could require 4 to 8 staff months of effort, depending on the complexity of the entity and the level of detail modeled to achieve an appropriate representation.

Addition of elements that are pervasive throughout the battlespace generally requires a larger level of effort than addition of battlespace entities. For example, introduction of weapons technologies that differ in kind, rather than degree, from conventional munitions currently represented in the STOW simulation would require some modification to virtually all elements of the battlespace. Such large efforts could take on the order of several or more staff years of programming and testing. Figure D-1 gives examples of the order of typical classes of modifications to the battlespace.

The level of effort associated with the KA required to support development varies by the same factors that affect code development time, but to a lesser degree. For new battlespace elements that are simply additional variants of entities that are already represented, the KA effort will typically take between 1 and 5 staff days, depending on the complexity of the entity and thus the amount of data required for code development. For new elements that are unique from anything already in the battlespace, the KA effort must first define the inputs, outputs, and controls associated with that entity and then collect the necessary data. The level of effort for KA for such elements could run from 2 to 4 staff weeks. For pervasive elements as discussed above, the level of effort for the supporting KA could take 1 to several staff months because of the necessity to address so many battlespace elements.

What is the level of effort required to add new battlespace conditions?

Addition of conditions represented through the existence or behavior of entities was discussed earlier. The other class of conditions that are represented in STOW simulations are physical conditions. The following discussion, therefore, will address the issues of adding physical conditions to the battlespace within STOW. The STOW program has categorized these physical conditions as terrain, environmental conditions, and dynamic terrain objects, which include elements such as buildings, bridges, obstacles, and craters.

Variable Costs: Development

		VERY EASY < 1 staff months	EASY < 6 staff months	HARD < 12 staff months	VERY HARD < 24 staff months
ENTITIES	Physical Models	+ small #'s of models + 2D/PVD view change parameters	+ new model w/ mod components + 3D view	+ new model w/new components + large #'s of models	pipeline functionality + GPS or IR + acoustics/fratricide
	Physical Models	task decomposition +small #'s of behaviors change parameters	modify tasks	+ new tasks	+ new architecture drills react to stimuli i.e. NBC
ENVIRONMENT	Terrain	use SWA	compile from WWTDB	———	+ small, hi-res DB 100x100kms
	Dynamic Terrain Objects	emplace objects	+ new objects, i.e. bridge w 4 damage states	+ interactive 3D(see thru hole in wall)	+ urban warfare (room-to-room)
	METOC	re-use weather DB	———	+ 7 days/season X 4 + real weather	develop new wx model
	Weather Effects	———	integrate existing model	———	develop new wx effects
	Dynamic Effects	———	———	+ contrails/wakes	———
C4I	Current Systems	Upgrade to new version of GCCS	+ new messages/ C4I system	———	+ two-way comms HITL & sim
	Future Systems	———	———	+ developmental/ conceptual C4I system	———
Other	ASSESSMENT	+ less than 5 data collection agents	+ less than 10 data collection agents Collect new data not currently emitted	———	———

Figure D-1. Approximate Level of Effort Required for Commonly Required Changes to the Battlespace

As with any other aspect of a simulation, addition of physical conditions has two major aspects—KA and code development. The KA required is not only the definition of the physical properties that must be represented, but also the level of detail required in their description. Required physical conditions and the level of detail needed in their representation are dependent on the battlespace elements that detect them, react to them, or are affected by them. The level of effort associated with the KA necessary for these conditions is therefore of the same order of magnitude as for the KA of the battlespace elements discussed above.

As can be expected, the level of effort associated with the software development and test needed for the representation of any given condition depends on the level of complexity of that condition. For terrain, processes are being developed that will make the production of terrain descriptions for any given area of the world at one specific level of detail a relatively simple task, requiring a few staff weeks to a couple of staff months of effort. More detailed data bases will require more effort. A data base as large and complex as that used for the STOW 97 demonstration, for example, would require as little as a few staff months if the

data is readily available to as much as a couple of staff years if extensive data acquisition and formatting are required.

The effort required to add new terrain objects also depends on complexity. For objects that can be created through the modification of existing objects, coding and testing can take as little as a staff week. For objects that require an entirely new development effort or require modification of the established data formats, up to several staff months of effort to develop, test, and incorporate into the STOW simulation system may be necessary.

Environmental factors are represented in the STOW simulation system as values in a spatially gridded database. To represent a variable not currently in the STOW environmental description (1) that cannot be derived from the values currently represented and (2) that must be represented with spatial variability could take a staff month or more. If spatial variability is not required or the new variable can be derived from existing variables, considerably less time may be required for its representation.

When the addition of any battlespace element or physical condition is required, consideration must be given to the need for a three-dimensional representation for the stealth viewers. Such views require appropriate models. Addition of three-dimensional models for elements can take as little as a few hours, if the model exists and must simply be imported into the system, to a couple weeks for the creation of complex models with many articulated parts. Representation of three-dimensional effects such as fog is image-generator specific. Because of this, more intricate code changes are required for their representation. The level of effort required for the addition of these representations is therefore more difficult to estimate, but generally falls in the range of 1 to a few staff months.

What is the level of effort required to add new battlespace behaviors?

The types of effort required to add new behaviors are the same as to add new battlespace elements. KA is required to define the behaviors at the proper level, identify the key aspects that must be represented and those that can be neglected, and present this data in a form appropriate for the software developers. Software development is then required to translate these representations in executable code that can be used in the simulation.

The level of effort required to add a behavior to the battlespace, like that of other elements, depends on the complexity of the behavior. Behaviors of individual entities are easier than those of groups or units. Behaviors that are more general in natural are more difficult than those that are specific. The greater the number of conditions, such as weather, diurnal effects, or force strength considerations, that must be accommodated by the behavior, the greater the level of effort required to incorporate the behavior into the simulation. The

level of effort associated with coding and testing a new behavior in JSAF task frames ranges from one to several staff months. The level of effort associated with adding a behavior with one of the representation methods such as artificial intelligence ranges from a few to several staff months.

KA associated with new behaviors is slightly more involved than that for a new battlespace element. Behaviors are first broken down into definable tasks. Each of these tasks is essentially an output to a physical model, so behavior decomposition must be linked to the representation of the physical entities. If all the tasks of a new behavior already exist as part of other behaviors, the KA for that behavior will be little more than task decomposition. In such cases, KA may take as little as a staff week for a new behavior and a few staff days for a modified behavior. If new tasks must be defined, the level of effort could increase to 2 to 4 staff weeks.

How easy is it to change programmed tactics and behaviors?

Tactics in JSAF are programmed as behaviors. Although a JSAF operator has several behaviors to choose from to accomplish a given mission, he does not have the flexibility to change any given behavior during run time. Changing behaviors and tactics requires code changes prior to simulation execution. Changing existing behaviors is typically easier than creating new ones, but the required steps are the same. The level of effort associated with changing a behavior depends on the complexity of the behavior and how significant the change. Minor changes may take only a small percentage of the effort required to originally produce the behavior. Major changes may be more easily and effectively accomplished by creating the behavior from scratch.

What level of effort is required to accommodate interactions with new entities?

Within the JSAF application, interactions are executed as behaviors. The level of effort required to add or modify interactions is determined using the same considerations discussed under the level of effort required to change tactics and behaviors.

What level of effort is required to add new interfaces to live or virtual systems?

Development of an interface requires that the protocol be defined, the format of the information exchanged be agreed upon, and the architecture of the interconnectivity be established. If, as is usually the case, no one person has detailed enough information to address these issues for both sides of the interface, then the efforts of an engineer from both associated systems is required to establish the interface. Typically, efforts to track changes, ensure configuration management, and coordinate the efforts of both systems are also

required. Thus, approximately 2½ full-time staff for some period of time are required to add a new interface. Depending on the complexity of the interface, the time required could range from 2 to several months for development and test of the desired interface.

In general, interfaces in which both sides are not controlled by one program are high-maintenance areas. Any change to either system could result in a change to the interface. Many of the systems to which STOW is interfacing, particularly the GCCS segments, are in a state of development and are thus changing frequently. Some interfaces are in a state of nearly continuous revision. The maintenance of the interfaces to the live systems, therefore, currently requires two full-time staff engineers. Increased use of the Defense Information Infrastructure (DII) Common Operating Environment (COE) should mitigate this effort considerably in the future. In addition, the implementation of the Java-based STOW C4I Gateway should reduce both development and maintenance times in the future.

What level of effort is required to interface to other constructive simulations?

As discussed before, there are several factors to consider in attempting to gain interoperability with another constructive simulation. The more similar the other simulation is to the STOW simulations in each of these respects, the easier achieving interoperability will be. The pervasiveness of the required interactions also affects the level of effort required to achieve interoperability. For example, a simulation that only needs to interoperate with one specific entity type would be less complex than a simulation that needs to interoperate with all entity types. In light of the many variables that can affect obtaining interoperability, estimating the required level of effort is extremely difficult. A minimum safe estimate for this effort, however, would be several staff months.

What level of effort is required to generate scenarios and instantiate units?

The activity key to supporting efficient scenario development is accurate definition of the event requirements. That is, the purpose for running the event and what data are required from it are needed to develop a suitable scenario. (Considerations for requirement definition are further addressed during evaluation.) The other factors that must be considered in scenario development are the desired force structure, the (simulated) terrain on which the event is to be conducted, limitations in the simulation, and availability of operators and processing resources.

Considering all the requirements and constraints, an initial scenario is developed. Participation of the ultimate customer of the data generated from the event (i.e., the analyst who will use the results) is critical to this development. After the initial scenario is developed, scenario files are created. Test runs are then conducted to ensure that the software

and resources can support the scenario as expected. These runs are also used to check the timing of coordinated operations of the synthetic forces and to identify any required adjustments. Production and capture of the data required to support the original intent of the event is also examined. Based upon the results of these tests, modifications are made to the scenario as required and the test process repeated.

The task of outlining a scenario can be done with a very small staff. This can be as few as one person if that person has all the required knowledge as cited above, but two or three persons is a more typical size. Use of a larger staff is acceptable and provides increased variety of input, but negatively affects efficiency as a larger web of communications must be established and maintained. Creation of scenario files typically requires on the order of a few staff days to a few staff weeks, depending on the size and complexity of the operations. Aircraft representations controlled by Soar artificial intelligence agents require some additional initial tasking effort, but once properly tasked, they are capable of much greater autonomy than other synthetic forces and require less operator intervention during execution. Conversion of an Air Tasking Order to commands programmed into the Soar aircraft typically require about a staff day of effort for each 80 to 100 sorties. Creation of the terrain object files is a straightforward task and requires less than a staff day. To ensure consistency among all data parameters, weather files are best created from historical records. Location of suitable weather files for the given location, time of year, and desired conditions during the event and ensuring that these files are properly formatted for incorporation into a STOW event can take from a couple of staff days to a few staff weeks.

Testing of the scenario files is done in stages. Small vignettes are tested first, then gradually combined until the entire scenario is represented. The number of cycles required to build to the entire scenario depends on its size and complexity. Initial vignette tests can be done independently with a minimum number of operators. The final scenario test, however, should be done with all the operators required for the event, as well as all the network connectivity and infrastructure in place and operating.

For any simulation event, there must always be some initial conditions and limitation of scope. Care must be taken that definition of either of these does not affect the final analysis or the data to support that analysis.

What level of effort is required for simulation control?

The number of operators required for control of the synthetic forces depends on the size of the synthetic forces, the complexity of the simulated operations, and the training level of the operators. Some rules of thumb, however, are approximately 5 operators for an Army

brigade; approximately 5 naval vessels per operator; and 1 operator for every 20 to 40 air missions to control re-tasking, assuming that proper air planning as discussed before is completed to allow for automated aircraft operation. If proper initialization of the aircraft was not performed, effective control of more than 8 to 10 aircraft per operator is not realistic. One operator is typically capable of monitoring and controlling all the aspects of the synthetic natural environment.

What level of effort is required to plan and prepare data collection and extract collected data?

The current scheme for data collection, extraction, and manipulation has only very recently been integrated into the STOW simulation system. This has two effects on the ability to confidently discuss the factors that determine the level of effort associated with data collection and extraction. First, there is not significant program experience with this new data collection and extraction process upon which to base estimates. Second, as the state of this integration is still somewhat immature, the levels-of-effort data that exist are heavily influenced by first-time integration efforts.

The effort to setup the AARS for the desired data reduction depends on the amount and type of reduction desired. The setup requires some programmer effort to develop the algorithms that control the population of collection agents and execute the desired data reduction. The participation of the analyst or subject matter expert in clearly delineating the mission statement, including the objective, the segments that constitute the mission, and criteria for mission completion, is critical. Such delineation is essential to ensure that the collection agents are properly developed the first time to collect the data necessary to address each test objective. Consider, for example, investigation of a new tactic that requires a certain minimum aircraft sortie rate. Suppose the sortie rate could not be met because of lack of sufficient fuel reserves at forward air bases, but that fuel reserve data was not collected. It would be known that the tactic failed because the sortie rate could not be sustained, but the reason it could not be sustained may not have been captured. This would preclude proper planning for correction of the shortcoming, which may result in the abandonment of a potentially valuable tactic. One to several days of interaction between the analyst and AARS programmer are required to ensure the programmer adequately understands the requirements for data extraction. Approximately the same amount of programmer time is required to complete coding of the collection agents and any further manipulation schemes

Run-time data collection within the STOW simulation system is done by logging all data packets exchanged between federates. The data-logging devices can be configured to

record only certain types of data such as transmissions at a particular frequency or interactions within a specific geographic region. If such filtering of recorded data is desired, a detailed and coordinated effort between the event designers and the analysts who will interpret the simulation results is required to ensure that the proper data is recorded and that no required data is lost. Similarly, given the possibility that the simulation supporting the test might crash, either globally or within a localized federate, close coordination between the technical teams supporting data collection (SLOGGER) and data translation (AARS) must be established from the outset. This will ensure crash recovery solutions (e.g., restart from a known save point) are compatible with data manipulation requirements and do not lead to wasted data collection efforts.

After data is recorded by the logger system, it must be properly configured for the AARS. As a rule of thumb, entry into the AARS repository requires an elapsed time equal to the actual time it took for the simulation to generate the data. That is, if analysis of a 4-hour segment of the simulation event is desired, it will take 4 hours to play this data from the logger into the AARS repository. During this transfer, standardized data collection agents populate the respective reporting fields to facilitate the desired data reduction.

Once the data collection effort has been stored in the AARS, further manipulation of the collected data can occur. The operator interested in specific data files can access the files from a password-controlled Internet site. Downloading data into Microsoft Excel files facilitates data manipulation. The data manipulation effort is not completely user friendly; that is, aggregating the data and subsequently coding the commands to ensure the aggregation occurs, still requires a trained analyst to assist the operator. Initiatives to develop translators to simplify the user interface requirements have been identified.

What level of effort is required to prepare for new events?

The activities required prior to an event are those of Phases I–III listed in Appendix A. Based upon the number of tasks involved and the potential variance in the level of effort associated with each of those tasks, it is possible that the level of effort required to prepare for an event can vary from a few staff months to several staff years.

To ensure that the preparation phase is accomplished in the most efficient means, emphasis must be placed on the early requirement definition activities. The event requirements must be developed in the context of the data required for the analysis of the candidate system and the capabilities of the simulation. Requirements that cannot be supported by the simulation should not be posed for a simulation event. Choosing requirements thus involves knowledge of both the candidate system under investigation and

the simulation system used for the investigation. Typically, the analyst associated with the system has the in-depth understanding of the desired purpose for running the event, but is not familiar with the capabilities of the simulation. Personnel associated with maintenance or operation of the simulation system understand the capabilities of the simulation, but are not familiar with the system under investigation and so do no have full insight into the purpose of the event. Education of all parties in the areas they lack knowledge is required to craft an investigation that can be executed within the limits of the data provided by the simulation. This education process and the coordination required to merge the two collections of knowledge is a principal challenge of the requirements definition phase.

Once some level of understanding is gained by both groups, the iterative process of defining the measures of effectiveness to be investigated through the experiment and the changes required in the STOW simulation system to support them can begin. Each iteration should add some level of detail, and these iterations need to continue until precise definitions of the data to be generated and collected and all changes to battlespace element representations are obtained. The more care taken in completing these definitions, the more efficient the execution of the subsequent efforts.

In most cases, the different tasks listed under the tool integration phase can be conducted in parallel. New entities could be added, for example, at the same time a new terrain database is being generated. The ranges of the levels of effort required for most of these tasks were presented above. The efforts for this phase of event preparation will often require the greatest expenditure of resources. In addition to those efforts, note that the greater the necessary changes in any or all of the simulation capabilities, the greater the associated integration effort. The event planning staff must ensure, therefore, that adequate integration and test effort is planned for.

A major focus of the event preparation phase is the creation and test of a scenario that will generate the required data. The major tasks associated with that activity were previously discussed. Additional tasks that must occur in this phase are to create site-specific hardware configurations, establish and test network connectivity, refine the MOEs, and finalize the test and data collection plans. The hardware site configuration and network connectivity tasks typically take about a staff week or less per site. The effort associated with the finalization of the MOEs and planning documents depends on the complexity of the designed test and the proficiency of the assigned staff. From a couple of staff weeks to a few staff months could be required for these efforts.

What level of effort is required to execute events?

The staffing required for event execution includes the synthetic forces operators, the synthetic environment operator, local area network (LAN) system administration at each site, wide area network administration, distributed event management (DEM) monitoring and control, administration of external system interfaces, and event direction and control.

The number of operators required for control of the synthetic forces and natural environment was discussed before. Generally, one systems administration person is required per site for LAN maintenance. Two controllers can administer the wide area network for networks with as many as 10 nodes. One operator for centralized DEM monitoring and control is prudent with site support from the LAN administrators needed. Management of external interfaces depends on the specific interfaces used, but generally one administrator for every two or three interfaces is sufficient to react to any system malfunctions. For each event, a decision on the method and extent of event direction and control needs to be made, but this effort should only rarely require more than one or two persons. For planning purposes, it is wise to have the entire execution team in place 2 days before the event is to commence to ensure there are no connectivity or application problems.

Is significant education on the system required to obtain satisfactory results?

Basic operator training is required if satisfactory results are to be obtained. Familiarization with the graphical user interface (GUI) is needed to understand the mouse controls, how to manipulate the display, and which windows are available. Operators require training in how to create and task units and what they look like when they are running. The basics of SAF operation can be learned within a week. Efficiency in monitoring and control increases with experience, as operators learn which displays contain the data required for monitoring various aspects of a unit's execution and behavior and to best provide additional or subsequent controls. One to two months of operator training should be planned before participation in his or her first event. As the SAF simulate military operations, efficient operation is enhanced if the SAF operator has some knowledge of such operations.

What effort is required to maintain the system?

Maintenance of the system infrastructure consists of ensuring proper configuration and operation of the processing hardware and local and wide area network equipment. Hardware maintenance for the site of the STOW facility in the Joint Training, Analysis, and Simulation Center (JTASC) averages less than a staff day per week, but depends on the activity level of the site. Additional effort is required to ensure that the software configuration for each piece of hardware is correct. In addition to the application software,

attention must be given to ensure that the operating system version, drivers, kernels, and protocol daemons are kept current and consistent. The level of effort associated with these activities is dependent on the level of change activity associated with the operating system federate and application federates. Most of the current change activity is associated with maturing the software. This cause for change activity should be expected to decrease over the next year. If the system is used to support assessment of several systems per year, however, the total change activity would probably not decrease and may actually increase. Planning on one full-time equivalent at each active simulation site for the maintenance system infrastructure is therefore prudent. The software for the network hardware, routers, and switches is commercial. Upgrades to this software are typically part of the maintenance agreement with the hardware provider. Prudence also dictates budgeting the cost of maintenance contracts for this commercial hardware and associated software into the cost of system operation. As new software releases for the switches and routers are received, the services of a wide area network expert is required to ensure that the new releases support the manner in which the STOW simulation system uses an asynchronous transfer method (ATM) network. A few staff days per quarter is expected to be necessary for such activities.

What level of effort is required to maintain the software?

Software maintenance can include improving or expanding current capabilities, as well as correctng latent defects (i.e., fixing bugs). The levels of effort required to expand the capabilities of the applications through additions to the battlespace were discussed above. The level of effort associated with bug fixes can very much be tailored to the resources available. If bugs are identified at a rate faster than affordable staffing can fix them, a backlog will develop. In such a case, having some process to catalog and establish the order in which defects will be fixed is imperative. Because of the recent changes to JSAF, it would probably require about 4 staff years to stabilize the software. Some of this effort will be applied over the next year, but some amount of stabilization should still be expected at the end of the ACTD residual phase. Because of this, the minimum number of software maintenance personnel recommended for the first few years after the residual phase is four. This will provide a capability to address both correction of latent defects and addition of new elements, conditions, and behaviors that may be required for desired analyses.

In addition to the activity of the software maintenance personnel, some level of management attention is required for effective configuration management. As a minimum, management personnel should supervise the prioritization of bug fixes and incorporation of enhancements.

Are any special facilities or expertise required for life-cycle maintenance?

Maintenance of the synthetic force and natural environment applications is done on the same machines as used for execution during an event. No special license agreements are needed for the execution or maintenance of any of the applications; therefore, no special facilities are required for life-cycle maintenance. Expertise in the software applications is obviously required for their maintenance. Each of these applications is quite complex. Developing the specific expertise required to efficiently maintain any one of the applications typically requires about 6 months, given the proper background in software coding in general and the particular language of the application in particular.

D.9 RELIABILITY AND AVAILABILITY

Does the simulation's reliability support timely and cost-effective generation of the required data?

The system is reasonably stable and can provide simulation reliability in excess of 95 percent. Most current instability arises primarily from requests by the event controllers for quick turn-around changes in features or capabilities.

Does the availability of the system support timely and cost-effective generation of the required data?

The availability of the STOW simulation system is dependent on the availability of the resources required to operate the system. These resources are both hardware and personnel. For the duration of the STOW program, the availability of any of these resources was never an issue. With the end of the STOW program, the disposition of the equipment used by the program and the future tasking of the personnel most experienced in the system's maintenance and use is in question. Most of the hardware used for STOW is PC based, so hardware availability should not be an issue for most potential users. Personnel availability may be a difficult problem in that as core personnel move to other programs, they may not be available for future employments of the STOW simulation system.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	March 2000	Final — July 1998 – September 1999

4. TITLE AND SUBTITLE
The Evolving Role of Analysis in Complex Military Decision-Making

5. FUNDING NUMBERS
DASW01 98 C 0067
DARPA Assignment DA-3-1697

6. AUTHOR(S)
Robert J. Graebener, Stephen Kaputis

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Institute for Defense Analyses
1801 N. Beauregard St.
Alexandria, VA 22311-1772

8. PERFORMING ORGANIZATION REPORT NUMBER
IDA Document D- 2415

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
DARPA/DISA JPO
4601 North Fairfax Drive
Suite 704
Arlington, VA 22203

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 180 words)*

The Synthetic Theater of War (STOW) Advanced Concept Technology Demonstration (ACTD) concluded its life cycle with USJFCOM (operational user) acceptance of the federated simulation in FY 99. During the ACTD life cycle, USJFCOM required the technologies to support the following: Joint task force training (Unified Endeavor 98-1); Analysis of ACTDs (Joint Counter Mine); and joint experimentation (J9901, Critical Mobile Target, Attack Operations) circa 2015. This report documents the complexities associated with current and future problem-solving techniques necessary to meet the decision-maker's needs in the 21st century. The STOW ACTD is a case study in what to expect from future challenges facing the warfighter and the analytical community pledged to support the joint commander. Using the traditional problem-solving approach as a starting point, this paper addresses the question: "How can the decision-maker of tomorrow employ the latest technologies and analytical processes to make the best decision?" The answer lies in the following: Establish standard procedures for decisions; ensure all stakeholders are included; identify long-lead items early; and establish early working relationships between the maintainers of the analysis tools and the users.

14. SUBJECT TERMS
ACTD, Advanced Concept Technology Demonstration, measures of effectiveness, MOE, STOW, Synthetic Theater of War

15. NUMBER OF PAGES
92

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR